Manual de Utilização
UCPs Série Nexto
NX3004, NX3010, NX3020, NX3030

Rev.J 10/2015

Cód. Doc.: MU214100

altus

www.altus.com.br

Nenhuma parte deste documento pode ser copiada ou reproduzida sem o consentimento prévio e por escrito da Altus Sistemas de Automação S.A., que se reserva o direito de efetuar alterações sem prévio comunicado.

Conforme o Código de Defesa do Consumidor vigente no Brasil, informamos, a seguir, aos clientes que utilizam nossos produtos aspectos relacionados com a segurança de pessoas e instalações.

Os equipamentos de automação industrial fabricados pela Altus são robustos e confiáveis devido ao rígido controle de qualidade a que são submetidos. No entanto, equipamentos eletrônicos de controle industrial (controladores programáveis, comandos numéricos, etc.) podem causar danos às máquinas ou processos por eles controlados em caso de defeito em suas partes e peças ou de erros de programação ou instalação, podendo inclusive colocar em risco vidas humanas.

O usuário deve analisar as possíveis consequências destes defeitos e providenciar instalações adicionais externas de segurança que, em caso de necessidade, sirvam para preservar a segurança do sistema, principalmente nos casos da instalação inicial e de testes.

Os equipamentos fabricados pela Altus não trazem riscos ambientais diretos, não emitindo nenhum tipo de poluente durante sua utilização. No entanto, no que se refere ao descarte dos equipamentos, é importante salientar que quaisquer componentes eletrônicos incorporados em produtos contêm materiais nocivos à natureza quando descartados de forma inadequada. Recomenda-se, portanto, que quando da inutilização deste tipo de produto, o mesmo seja encaminhado para usinas de reciclagem que deem o devido tratamento para os resíduos.

É imprescindível a leitura completa dos manuais e/ou características técnicas do produto antes da instalação ou utilização do mesmo.

Os exemplos e figuras deste documento são apresentados apenas para fins ilustrativos. Devido às possíveis atualizações e melhorias que os produtos possam incorrer, a Altus não assume a responsabilidade pelo uso destes exemplos e figuras em aplicações reais. Os mesmos devem ser utilizados apenas para auxiliar na familiarização e treinamento do usuário com os produtos e suas características.

A Altus garante os seus equipamentos conforme descrito nas Condições Gerais de Fornecimento, anexada às propostas comerciais.

A Altus garante que seus equipamentos funcionam de acordo com as descrições contidas explicitamente em seus manuais e/ou características técnicas, não garantindo a satisfação de algum tipo particular de aplicação dos equipamentos.

A Altus desconsiderará qualquer outra garantia, direta ou implícita, principalmente quando se tratar de fornecimento de terceiros.

Os pedidos de informações adicionais sobre o fornecimento e/ou características dos equipamentos e serviços Altus devem ser feitos por escrito. A Altus não se responsabiliza por informações fornecidas sobre seus equipamentos sem registro formal.

DIREITOS AUTORAIS

Nexto, Série Ponto, MasterTool, Quark, ALNET e WebPLC são marcas registradas da Altus Sistemas de Automação S.A.

Windows, Windows NT e *Windows Vista* são marcas registradas da Microsoft Corporation.

Este(s) produto(s) utiliza(e) tecnologia EtherCAT® (www.ethercat.org).

NOTIFICAÇÃO DE USO DE SOFTWARE ABERTO

Para obter o código fonte de componentes de software contidos neste produto que estejam sob licença GPL, LGPL, MPL, entre outras, favor entrar em contato através do e-mail opensource@altus.com.br. Adicionalmente ao código fonte, todos os termos da licença, condições de garantia e informações sobre direitos autorais podem ser disponibilizadas sob requisição.

Sumário

1. INTRODUÇÃO 7

Série Nexto 8
Características Inovadoras 9
Documentos Relacionados a este Manual 10
Inspeção Visual 11
Suporte Técnico 11
Mensagens de Advertência Utilizadas neste Manual 11

2. DESCRIÇÃO TÉCNICA 13

Painéis e Conexões 13
Características Gerais 14
Características Comuns 14
Características Específicas 16
Interfaces Seriais 20
Interfaces Ethernet 21
Fonte de Alimentação 22
Interface do Cartão de Memória 23
Compatibilidade com Demais Produtos 23
Desempenho 23
Tempos de Aplicação 24
Tempos para Execução de Instruções 24
Tempos de Inicialização 24
Tempo de Intervalo 24
Dimensões Físicas 25
NX3004 25
NX3010, NX3020 e NX3030 26
Dados para Compra 27
Itens Integrantes 27
Código do Produto 27
Produtos Relacionados 27

3. INSTALAÇÃO 29

Instalação Mecânica 29
Instalação Elétrica 29
NX3004 29
NX3010, NX3020 e NX3030 30
Conexão com a Rede Ethernet 31
Endereço IP 31
ARP Gratuito 32
Instalação do Cabo de Rede 32
Conexão com a Rede Serial RS-232C 33
Comunicação RS-232C 34
Conexão com a Rede Serial RS-485/422 34
Comunicação RS-485 sem Terminação 35
Comunicação RS-485 com Terminação Interna 36
Comunicação RS-485 com Terminação Externa 36
Exemplo de Ligação de Rede RS-485 com Terminação Externa e Redundância de Mestre 37
Comunicação RS-422 sem Terminação 38
Comunicação RS-422 com Terminação Interna 39

Comunicação RS-422 com Terminação Externa 40
Exemplo de Rede RS-422 41
Instalação do Cartão de Memória 41
Instalação da Arquitetura 42
Instalação de Módulos no Bastidor Principal 42
Instalação do Programador 43

4. CONFIGURAÇÃO 44

Configuração da UCP 44
Parâmetros Gerais 44
Configuração Evento Externo 54
Configuração do SOE 56
Sincronização de Tempo 59
Configuração das Interfaces Seriais 61
COM 1 (NX3010/NX3020/NX3030) 61
COM 1 (NX3004) e COM 2 (NX3010/NX3020/NX3030) 64
Configuração das Interfaces Ethernet 65
Interfaces Ethernet Locais 65
Interfaces Ethernet Remotas 66
Portas TCP Reservadas 66
Configuração do Módulo NX5000 66
Configuração de Protocolos 68
Comportamento dos Protocolos x Estados da UCP 71
MODBUS RTU Mestre 72
MODBUS RTU Escravo 85
MODBUS Ethernet 93
OPC DA 116
EtherCAT 129
Desempenho de Comunicação 149
Taxa de Comunicação de um Dispositivo MODBUS Servidor 149
Taxa de Comunicação de um Dispositivo com Servidor OPC 151
Disparo de Relações MODBUS Cliente de Forma Acíclica 152
Desempenho do Sistema 152
Escaneamento de E/S 152
Cartão de Memória 153
Relógio RTC 153
Blocos Funcionais e Funções para Leitura e Escrita do RTC 154
Estruturas de Dados do RTC 158
Memória de Arquivos de Usuário 160
Cartão de Memória 162
Acesso no MasterTool 164
Menu Informativo e de Configuração da UCP 165
Blocos Funcionais e Funções 167
Blocos Funcionais Especiais para Comunicação Serial 167
Atualização de Entradas e Saídas 183
Bloco Funcional PID 186
Timer Retentivo 188
Timer Não-Redundante 192
Log de Usuário 194
SNMP 197
Introdução 197
SNMP nas UCPs Nexto 198
Private MIB 199
Configuração 202
Usuário e Comunidades SNMP 204

Gerenciamento de Usuários e Direitos de Acesso **204**

5. PROGRAMAÇÃO INICIAL **205**

Organização e Acesso à Memória **205**
Perfis de Projeto **207**
Simples 208
Básico 208
Normal 208
Experiente 209
Personalizado 210
Perfil de Máquina 210
Tabela Geral 211
Número Máximo de Tarefas 211
Configurando a UCP **213**
Bibliotecas **214**
Inserindo uma Instância de Protocolo **214**
MODBUS RTU 214
MODBUS Ethernet 216
Localizando a Rede **218**
Login **220**
Modo Run **221**
Modo Stop **223**
Escrita e Forçamento de Variáveis **223**
Logout **224**
Upload do Projeto **224**
Estados de Operação da UCP **226**
Run 226
Stop 226
Breakpoint 226
Exception 226
Reset a Quente 226
Reset a Frio 226
Reset Origem 226

6. REDUNDÂNCIA COM UCP NX3030 ..... **228**

Introdução **228**
Descrição Técnica e Configuração **230**
Configuração Mínima de um CP Redundante (Sem utilização do Painel PX2612) 230
Configurações Típicas de um CP Redundante 230
Módulo NX4010 231
Painel de Controle de Redundância PX2612 232
Interconexões entre Half-Clusters e Painel de Controle de Redundância PX2612 234
Características Gerais 235
Dados para Compra 237
Princípios de Funcionamento **238**
Identificação de uma UCP NX3030 238
Projeto Redundante Único 238
Estrutura do Projeto Redundante 238
Mapeamentos Múltiplos 243
Estruturas de Dados de Diagnósticos, Comandos e Usuário 244
Serviços de Sincronização Cíclicos através de NETA e NETB 245
Serviços de Sincronização Esporádicos através de NETA e NETB 246
Desabilitação da Sincronização de Projetos 247
Configurações de Redes PROFIBUS 248

Redes Ethernet Redundantes com NIC Teaming 249
Métodos de Troca de IP 249
Uso Combinado de NIC Teaming e IP Ativo 253
Uso de Interfaces Ethernet com Indicação de Falha Vital 253
Uso de Comunicação OPC com Projetos Redundantes 254
Estados de um CP Redundante 254
Funções do Painel de Comando de Redundância PX2612 256
Transições entre Estados de Redundância 258
Primeiros Instantes em Estado Ativo 261
Falhas mais Comuns Causadoras de Switchovers Automáticos entre Half-Clusters 262
Falhas Associadas a Switchovers entre Half-Clusters Gerenciados pelo Usuário 262
Tolerância a Falhas 263
Overhead da Redundância 265
Programação de um CP Redundante 266
Wizard para Criação de um Novo Projeto Redundante 266
Configuração dos Half-Clusters 270
Configurações das Portas Ethernet da UCP NX3030 (NET 1 e NET 2) 270
Configurações dos Módulos NX5001 272
Configurações dos Módulos NX5000 274
Configurações do Módulo NX4010 275
Configurações de I/O Drivers 276
Configurações da MainTask 276
Objeto Redundancy Configuration 278
GVL Diagnostics 278
GVLs com Variáveis Simbólicas Redundantes 279
POUs do Tipo Program com Variáveis Simbólicas Redundantes 279
Utilização de Breakpoints em Sistemas Redundantes 279
Gerenciamento de Instâncias MODBUS em Sistemas Redundantes 279
Limitações na Programação de um CP Redundante 280
Obtendo o Estado da Redundância de um Half-Cluster 280
Leitura de Diagnósticos Não-Redundantes 281
Carga de Programas em um CP Redundante 281
Carga Inicial de um Projeto Redundante 282
Conexão do MasterTool com uma UCP NX3030 de um CP Redundante 284
Carga de Modificações em um Projeto Redundante 285
Carga de Modificações Off-Line e On-Line 285
Carga On-Line de Modificações 286
Carga Off-Line de Modificações com Interrupção do Controle do Processo 287
Planejamento Prévio para Modificações Off-Line sem Interrupção do Controle do Processo 288
Explorando a Redundância para Carga Off-Line de Modificações sem Interrupção do Controle do Processo 293
Manutenção 297
Troca a Quente de Módulos em um CP Redundante 297
Mensagens de Advertência do MasterTool 297
Diagnósticos da Redundância no Visor Gráfico da UCP NX3030 298
Estrutura de Diagnósticos da Redundância 298
Teste do Painel PX2612 309

7. MANUTENÇÃO 312

Diagnósticos do Módulo 312
One Touch Diag 312
Diagnósticos via LED 315
Diagnósticos via WEB 316
Diagnostic Explorer 319
Diagnósticos via Variáveis 319

Diagnósticos via Blocos Funcionais 334
Visor Gráfico 335
Log de Sistema 337
Não Carregar a Aplicação na Inicialização 338
Falha na Alimentação 338
Problemas mais Comuns 338
Solução de Problemas 339
Manutenção Preventiva 339

8. GLOSSÁRIO 340

ANEXO A. INTEROPERABILIDADE DNP3 343

DNP3 Device Profile 343
DNP3 343
DEVICE PROFILE DOCUMENT 343
DNP V3.0 Implementation Table 344

1. Introdução

As UCPs da Série Nexto foram desenvolvidas para atender as necessidades de diversos clientes nas mais variadas aplicações presentes na automação industrial e no controle de processos. Devido ao formato compacto e robusto, excelente desempenho de processamento e atualização rápida das E/S, proveniente de um único barramento de comunicação de alta velocidade, as UCPs da Série Nexto são as melhores escolhas para as mais exigentes necessidades de aplicação de controle. Em complexas aplicações, onde a confiabilidade, disponibilidade e operação remota de E/S são necessárias, as UCPs da Série Nexto são igualmente uma grande opção, devido às diferentes topologias de redundância e possibilidades de expansão de bastidores.

As UCPs da Série Nexto são providas de um inovador e único serviço de diagnósticos, oferecendo ao usuário uma experiência totalmente nova. Utilizando apenas uma tecla localizada na parte superior do módulo, embutido ao compacto visor gráfico, o usuário tem acesso direto a muitas informações sobre os módulos de E/S, interfaces de redes de campo e muitos outros módulos da aplicação. Além disso, possui serviços de sistema de registro de usuários e facilidade de depuração e gerenciamento de tarefas, reduzindo o custo da aplicação e tempo de instalação.

Finalmente, as UCPs Série Nexto apresentam diversas interfaces de comunicação, como portas seriais e portas Ethernet, interface com cartão de memória e linguagens de programação conforme norma IEC 61131-3.

Figura 1-1. UCP NX3030

Série Nexto

A Série Nexto é uma poderosa e completa série de Controladores Programáveis (CP) com características exclusivas e inovadoras. Devido a sua flexibilidade, design funcional, recursos de diagnóstico avançado e arquitetura modular, o CP Nexto pode ser usado para controle de sistemas em aplicações de pequeno, médio ou grande porte.

A arquitetura da Série Nexto possui uma extensa variedade de módulos de entradas e saídas. Estes módulos combinados com um poderoso processador de 32 bits e um barramento de alta velocidade baseado em Ethernet se adéquam a inúmeros tipos de aplicações como controle de alta velocidade para máquinas de pequeno porte, complexos processos distribuídos, aplicações redundantes e sistemas com grande número de E/S. Além disto, a Série Nexto possui módulos de comunicações com as mais populares redes de campo, entre outras características.

A Série Nexto possui uma avançada tecnologia em seu barramento que utiliza uma interface Ethernet determinística de alta velocidade, possibilitando que informações de entradas, saídas e dados possam ser compartilhados entre os módulos do sistema com máxima eficiência. O sistema pode ser facilmente distribuído em campo, possibilitando o uso de expansões de bastidores com o mesmo desempenho de um módulo local permitindo que todos os tipos de módulos sejam utilizados tanto no bastidor local quanto nas expansões de bastidores sem restrições. Para a interligação entre as expansões de bastidores é utilizado um simples cabo padrão Ethernet.

Figura 1-2. Série Nexto – Visão Geral

Características Inovadoras

A Série Nexto traz aos usuários diversas inovações na utilização, supervisão e manutenção do sistema. Estas características foram desenvolvidas focando um novo conceito em automação industrial. A lista abaixo mostra algumas destas características que o usuário encontrará na Série Nexto:

Battery Free Operation: A Série Nexto não requer nenhum tipo de bateria para manutenção de memória e operação de relógio de tempo real. Esta funcionalidade é extremamente importante porque reduz a necessidade de manutenção do sistema e permite o uso em locais remotos de difícil manutenção. Além disto, esta característica é ambientalmente correta.

Multiple Block Storage: Diversos tipos de memória estão disponíveis nas UCPs da Série Nexto oferecendo a melhor opção para cada necessidade. Estas memórias são divididas em memórias voláteis e memórias não voláteis. Para uso de memórias voláteis as UCPs da Série Nexto oferecem variáveis de entrada de representação direta (%I), variáveis de saída de representação direta (%Q), variáveis de memória de representação direta (%M), memória de dados e memória de dados redundantes. Para aplicações que necessitam funcionalidades de memória não volátil, a Série Nexto possibilita a utilização de variáveis de representação direta de memória retentiva (%Q), memória retentiva de dados, memória de representação direta de memória persistente (%Q), memória persistente de dados, memória de programa, memória de código fonte, sistema de arquivo na UCP (DOC, PDF, dados) e interface para cartão de memória.

One Touch Diag: Esta é uma característica exclusiva dos CPs da Série Nexto. Através deste novo conceito o usuário pode checar as informações de diagnóstico de qualquer módulo do sistema diretamente no visor gráfico da UCP mediante apenas um pressionamento no botão de diagnóstico do respectivo módulo. A OTD é uma poderosa ferramenta de diagnóstico que pode ser usada offline (sem supervisor ou programador) e reduz os tempos de manutenção e comissionamento.

OFD – On Board Full Documentation: As UCPs da Série Nexto tem a capacidade de armazenar a documentação completa do projeto na sua memória. Este é um recurso interessante para fins de backup e manutenção já que a informação completa fica armazenada em um único e seguro local.

ETD – Electronic Tag on Display: Outra característica exclusiva apresentada pela Série Nexto é o ETD. Esta nova funcionalidade torna o processo de verificação do tag de qualquer ponto ou módulo de E/S usados no sistema diretamente no visor gráfico das UCPs. Juntamente com esta informação o usuário pode também verificar a descrição. Este é um recurso extremamente útil durante a manutenção e resolução de problemas.

DHW – Double Hardware Width: Os módulos da Série Nexto foram projetados para economizar espaço em painéis e nas máquinas. Por esta razão, a Série Nexto oferece duas diferentes larguras de módulos: largura dupla (com ocupação de 2 posições do bastidor) e largura simples (com ocupação de 1 posição do bastidor). Este conceito permite o uso de módulos de E/S compactos com alta densidade de pontos de E/S juntamente com módulos complexos, como UCPs, mestres de rede de campo e módulos de fonte de alimentação.

UCP de Alta Velocidade: Todas as UCPs desta Série Nexto foram concebidas para fornecer ao usuário um excelente desempenho e atender a uma ampla gama de exigências nas aplicações. Por exemplo: as UCPs NX3010, NX3020 e NX3030 podem executar instruções de adição, multiplicação e subtração em menos de 15 ns para valores de tipo inteiro e em menos de 23 ns para valores de tipo real. Elas são igualmente capazes de executar 1000 laços PIDs em menos de 5 ms.

iF Product Design Award 2012: A Série Nexto foi vencedora do iF Product Design Award 2012 no grupo industry + skilled trades. Este prêmio é reconhecido internacionalmente como um selo de excelência e qualidade, considerado o Oscar do design na Europa.

Documentos Relacionados a este Manual

Para obter informações adicionais sobre a Série Nexto, podem ser consultados outros documentos (manuais e características técnicas) além deste. Estes documentos encontram-se disponíveis em sua última revisão em http://www.altus.com.br.

Cada produto possui um documento denominado Característica Técnica (CT), onde encontram-se as características do produto em questão. Adicionalmente o produto pode possuir Manuais de Utilização (os códigos dos manuais são citados na CT).

Por exemplo, o módulo NX1001 tem todas as informações de características de utilização e de compra na sua CT. Por outro lado, o NX5001 possui, além da CT, um manual de utilização (MU).

Aconselham-se os seguintes documentos como fonte de informação adicional:

Código	Descrição	Idioma
CE114000	Nexto Series – Technical Characteristics	Inglês
CT114000	Série Nexto – Características Técnicas	Português
CS114000	Serie Nexto – Especificaciones y Configuraciones	Espanhol
CE114700	Nexto Series Backplane Racks Technical Characteristics	Inglês
CT114700	Características Técnicas dos Bastidores da Série Nexto	Português
CS114700	Características Técnicas de los Bastidores de la Serie Nexto	Espanhol
CE114900	NX4010 Redundancy Link Module Technical Characteristics	Inglês
CT114900	Características Técnicas do Módulo de Redundância NX4010	Português
CS114900	Características Técnicas del Módulo de Redundancia NX4010	Espanhol
CE114902	NX5001 PROFIBUS-DP Master Technical Characteristics	Inglês
CT114902	Características Técnicas do Mestre PROFIBUS DP NX5001	Português
CS114902	Especificaciones y Configuraciones Maestro PROFIBUS-DP NX5001	Espanhol
CE114908	NX5110 e NX5210 PROFIBUS-DP Heads Technical Characteristics	Inglês
CT114908	Características Técnicas Interfaces Cabeça PROFIBUS-DP NX5110 e NX5210	Português
CS114908	Especificaciones y Configuraciones PROFIBUS-DP Interfaz Cabezas NX5110 e NX5210	Espanhol
CE114903	Ethernet Module NX5000 Technical Characteristics	Inglês
CT114903	Características Técnicas do Módulo Ethernet NX5000	Português
CS114903	Especificaciones y Configuraciones Modulo Ethernet NX5000	Espanhol
CT112500	Características Técnicas do Painel de Controle de Redundância PX2612	Português
MU214600	Nexto Series User Manual	Inglês
MU214000	Manual de Utilização Série Nexto	Português
MU214300	Manual del Usuario Serie Nexto	Espanhol
MU214605	Nexto Series CPUs User Manual	Inglês
MU214100	Manual de Utilização UCPs Série Nexto	Português
MU214305	Manual del Usuario UCPs Serie Nexto	Espanhol
MU299609	MasterTool IEC XE User Manual	Inglês
MU299048	Manual de Utilização MasterTool IEC XE	Português
MU299800	Manual del Usuario MasterTool IEC XE	Espanhol
MP399609	MasterTool IEC XE Programming Manual	Inglês
MP399048	Manual de Programação MasterTool IEC XE	Português
MP399800	Manual de Programación MasterTool IEC XE	Espanhol
MU214601	NX5001 PROFIBUS DP Master User Manual	Inglês
MU214001	Manual de Utilização Mestre PROFIBUS DP NX5001	Português
MU214301	Manual del Usuario Maestro PROFIBUS DP NX5001	Espanhol
MU214608	Nexto PROFIBUS-DP Head User Manual	Inglês
MU214108	Manual de Utilização da Cabeça PROFIBUS-DP Nexto	Português
MU214308	Manual de Utilización de la Cabeza PROFIBUS-DP Nexto	Espanhol
MU219000	Ponto Series Utilization Manual	Inglês

MU209000	Manual de Utilização da Série Ponto	Português
MU209508	Manual de Utilização Cabeça PROFIBUS PO5063V1 e Cabeça Redundante PROFIBUS PO5063V5	Português
MU219511	PO5064 PROFIBUS Head and PO5065 Redundant PROFIBUS Head Utilization Manual	Inglês
MU209511	Manual de Utilização Cabeça PROFIBUS PO5064 e Cabeça Redundante PROFIBUS PO5065	Português
MU209020	Manual de Utilização Rede HART sobre PROFIBUS	Português

Tabela 1-1. Documentos Relacionados

Inspeção Visual

Antes de proceder à instalação, é recomendável fazer uma inspeção visual cuidadosa dos equipamentos, verificando se não há danos causados pelo transporte. Verifique se todos os componentes de seu pedido estão em perfeito estado. Em caso de defeitos, informe a companhia transportadora e a Altus.

CUIDADO:
Antes de retirar os módulos da embalagem, é importante descarregar eventuais potenciais estáticos acumulados no corpo. Para isso, toque (com as mãos nuas) em uma superfície metálica aterrada qualquer antes de manipular os módulos. Tal procedimento garante que os níveis de eletricidade estática suportados pelo módulo não serão ultrapassados.

É importante registrar o número de série de cada equipamento recebido, bem como as revisões de software, caso existentes. Essas informações serão necessárias, caso se necessite contatar o Suporte Técnico da Altus.

Suporte Técnico

Para entrar em contato com o Suporte Técnico da Altus em São Leopoldo, RS, ligue para +55 51 3589-9500. Para conhecer os centros de Suporte Técnico da Altus existentes em outras localidades, consulte nosso site (http://www.altus.com.br) ou envie um e-mail para altus@altus.com.br.

Se o equipamento já estiver instalado, tenha em mãos as seguintes informações ao solicitar assistência:

- Os modelos dos equipamentos utilizados e a configuração do sistema instalado.
- O número de série da UCP.
- A revisão do equipamento e a versão do software executivo, constantes na etiqueta afixada na lateral do produto.
- Informações sobre o modo de operação da UCP, obtidas através do MasterTool IEC XE.
- O conteúdo do programa aplicativo, obtido através do programador MasterTool IEC XE.
- A versão do programador utilizado.

Mensagens de Advertência Utilizadas neste Manual

Neste manual, as mensagens de advertência apresentarão os seguintes formatos e significados:

PERIGO:
Relatam causas potenciais, que se não observadas, levam a danos à integridade física e saúde, patrimônio, meio ambiente e perda da produção.

CUIDADO:
Relatam detalhes de configuração, aplicação e instalação que devem ser seguidos para evitar condições que possam levar a falha do sistema e suas consequências relacionadas.

ATENÇÃO:
Indicam detalhes importantes de configuração, aplicação ou instalação para obtenção do máximo desempenho operacional do sistema.

2. Descrição Técnica

Este capítulo apresenta todas as características técnicas das UCPs da Série Nexto NX3004, NX3010, NX3020 e NX3030.

Painéis e Conexões

A figura a seguir mostra o painel frontal da UCP NX3030.

Figura 2-1. UCP NX3030

Como se pode observar na figura, na parte superior do painel frontal se encontra o visor gráfico utilizado para mostrar o estado e diagnósticos de todo o sistema, incluindo os diagnósticos específicos de cada módulo. O visor gráfico também oferece um menu fácil de usar que traz ao usuário um modo rápido para ler ou definir alguns parâmetros como: temperatura interna (somente leitura), contraste do visor gráfico, endereço IP para cada interface de rede (somente leitura) e hora local (somente leitura).

Logo abaixo do visor gráfico, encontram-se os 2 LEDs que indicam a ocorrência de diagnóstico e do circuito cão-de-guarda. A Tabela 2-1 mostra a descrição dos LEDs. Para maiores informações sobre os estados e significados dos LEDs, consultar o capítulo Diagnósticos via LED.

LED	Descrição
DG	LED de Diagnóstico
WD	LED de Cão-de-Guarda

Tabela 2-1. Descrição dos LEDs

As UCPs da Série Nexto contam com duas teclas disponíveis ao usuário. A Tabela 2-2 mostra a descrição das teclas. Para maiores informações sobre o botão de diagnóstico, consulte os capítulos

One Touch Diag e Menu Informativo e de Configuração da UCP. Para obter mais informações sobre a tecla MS, consulte o capítulo de Configuração – Cartão de Memória.

Teclas	Descrição
Botão de Diagnóstico	Tecla situada na parte superior do módulo. Utilizada para visualização dos diagnósticos no visor gráfico ou para navegação no menu informativo e de configurações da UCP.
MS	Tecla localizada no painel frontal. Utilizada para remover o cartão de memória com segurança.

Tabela 2-2. Descrição das Teclas

No painel frontal estão disponíveis as interfaces de conexão das UCPs da Série Nexto. Essas interfaces são de comunicação Ethernet, comunicação serial e a interface do cartão de memória. A Tabela 2-3 apresenta uma breve descrição dessas interfaces.

Interfaces	Disponível nos modelos	Descrição
NET 1	NX3004 NX3010 NX3020 NX3030	Conector tipo RJ45 de comunicação no padrão 10/100Base-TX. Permite a comunicação ponto a ponto ou em rede nos protocolos abertos, MODBUS TCP cliente e servidor, MODBUS RTU via TCP cliente e servidor. Para obter mais informações sobre utilização, consulte o capítulo Configuração – Configuração das Interfaces .
NET 2	NX3020 NX3030	Conector tipo RJ45 de comunicação no padrão 10/100Base-TX. Permite a comunicação ponto a ponto ou em rede nos protocolos abertos, MODBUS TCP cliente e servidor, MODBUS RTU via TCP cliente e servidor. Para obter mais informações sobre utilização, consulte o capítulo Configuração – Configuração das Interfaces .
COM 1	NX3010 NX3020 NX3030	Conector tipo DB9 fêmea para comunicação no padrão RS-232C. Permite a comunicação ponto a ponto ou em rede nos protocolos abertos, MODBUS RTU escravo ou MODBUS RTU mestre. Para obter mais informações sobre utilização, consulte o capítulo Configuração - Configuração das Interfaces Seriais Configuração.
COM 1	NX3004	Conector tipo DB9 fêmea para comunicação nos padrões RS-485 e RS-422. Permite a comunicação ponto a ponto ou em rede nos protocolos abertos, MODBUS RTU escravo ou MODBUS RTU mestre. Para obter mais informações sobre utilização, consulte o capítulo Configuração - Configuração das Interfaces Seriais.
COM 2	NX3010 NX3020 NX3030	
Fonte de Alimentação	NX3004	Conector de seis terminais com fixação. Permite alimentar os módulos da série Nexto conectados ao mesmo bastidor, fornecendo uma potência de 15 W.
MEMORY SLOT	NX3010 NX3020 NX3030	Conector para cartão de memória. Permite utilizar um cartão de memória para diferentes tipos de armazenamento de dados como: logs de usuário, páginas Web, a documentação do projeto e arquivos de origem. Para obter mais informações sobre utilização, consulte o capítulo Configuração – Cartão de Memória.

Tabela 2-3. Interfaces de Conexão

Características Gerais

Características Comuns

	NX3004, NX3010, NX3020, NX3030
Ocupação do bastidor	2 posições sequenciais
Linguagens de programação	Lista de Instruções (IL) Texto Estruturado (ST) Diagrama Ladder (LD) Sequenciamento Gráfico de Funções (SFC)

	Diagrama de Blocos Funcionais (FBD) Gráfico Funcional Contínuo (CFC)
Tipos de tarefas	Cíclica (periódica) Disparada por evento (evento de software) Disparado por evento externo (evento de hardware) Contínua (execução livre) Disparada por estado (evento de software)
Alterações online	Sim
Suporte a troca a quente	Sim
Suporte a redundância de barramento de expansão	Sim
Interfaces seriais	NX3004 – COM 1: 1 x RS-485 / RS-422 NX3010/NX3020/NX3030 – COM 1: 1 x RS-232C NX3010/NX3020/NX3030 – COM 2: 1 x RS-485 / RS-422
Protocolo MODBUS	Mestre e escravo RTU (COM 1 e COM 2) Cliente e servidor TCP (NET 1 e NET 2) Cliente e servidor RTU via TCP (NET 1 e NET 2)
Protocolo OPC	Sim
Protocolo EtherCAT	Sim, NX3020 e NX3030
Protocolo SNMP	Sim, versões v1, v2c e v3
Relógio de tempo real (RTC)	Sim Resolução de 1 ms e variação máxima de 2 s por dia
Cão-de-guarda	Sim
Indicação de status e diagnóstico	Visor gráfico, LEDs, páginas web e memória interna da UCP
One Touch Diag (OTD)	Sim
Electronic Tag on Display (ETD)	Sim
Isolação	
Lógica para terra de proteção ⏚	1250 Vac / 1 minuto
Lógica para interfaces Ethernet	1500 Vac / 1 minuto
Lógica para porta serial (COM 2)	1000 Vac / 1 minuto
Lógica para porta serial na NX3004 (COM 1)	1000 Vac / 1 minuto
Interfaces Ethernet para terra de proteção ⏚	1250 Vac / 1 minuto
Interfaces Ethernet para porta serial (COM 2)	1500 Vac / 1 minuto
Interfaces Ethernet para porta serial na NX3004 (COM 1)	1500 Vac / 1 minuto
Interface Ethernet para Interface Ethernet	1500 Vac / 1 minuto
Porta Serial (COM 2) para terra de proteção ⏚	1250 Vac / 1 minuto
Porta Serial na NX3004 (COM 1) para terra de proteção ⏚	1000 Vac / 1 minuto
Temperatura de operação	0 a 60 °C
Temperatura de armazenamento	-25 a 75 °C
Umidade relativa de operação e armazenamento	5 a 96 %, sem condensação
Revestimento de circuitos eletrônicos	Sim
Índice de proteção	IP 20

Normas	IEC 61131-2 IEC 61131-3 CE, diretivas de Compatibilidade Eletromagnética (EMC) e Dispositivos de Baixa Tensão (Low-Voltage Directive – LVD) CE RoHS 2002/95/EC
Dimensões do módulo (L x A x P)	36,00 x 114,63 x 115,30 mm
Dimensões da embalagem	44,00 x 122,00 x 147,00 mm
Peso	350 g
Peso com embalagem	400 g

Tabela 2-4. Características Comuns

Notas:

Tipos de tarefas: Tarefa é um objeto usado para chamar POUs. Uma tarefa pode ser disparada por período, eventos ou pode ainda ser executada no modo contínuo. Cada tarefa pode chamar uma ou mais POUs.

Relógio de tempo real (RTC): O tempo de retenção, tempo em que o relógio de tempo real continuará a atualizar a data e hora após a desenergização da UCP, é 15 dias para operação a 25 °C. Na temperatura máxima do produto o tempo de retenção é reduzido para 10 dias

Isolação: O termo lógica é usado para referenciar os circuitos internos como processador, memória e interfaces com o barramento.

Revestimento de circuitos eletrônicos: O revestimento de circuitos eletrônicos protege as partes internas do produto contra umidade, poeira e outros elementos agressivos a circuitos eletrônicos.

Características Específicas

	NX3004	NX3010	NX3020	NX3030
Memória de variáveis de entrada de representação direta (%I)	32 Kbytes	32 Kbytes	64 Kbytes	96 Kbytes
Memória de variáveis de saída de representação direta (%Q	32 Kbytes	32 Kbytes	64 Kbytes	96 Kbytes
Memória de variáveis de representação direta (%M)	16 Kbytes	16 Kbytes	32 Kbytes	64 Kbytes
Memória de variáveis simbólicas	2 Mbytes	4 Mbytes	5 Mbytes	6 Mbytes
Quantidade máxima de memória configurável como retentiva ou persistente	7,5 Kbytes	64 Kbytes	112 Kbytes	112 Kbytes
Memória de dados redundantes total	-	-	-	736 Kbytes
Memória de variáveis de entrada de representação direta (%I)	-	-	-	80 Kbytes
Memória de variáveis de saída de representação direta (%Q)	-	-	-	80 Kbytes
Memória de variáveis de representação direta (%M)	-	-	-	64 Kbytes
Memória de variáveis simbólicas	-	-	-	512 Kbytes
Memória de programa	3 Mbytes	4 Mbytes	6 Mbytes	8 Mbytes
Memória de código fonte (backup)	32 Mbytes	40 Mbytes	80 Mbytes	120 Mbytes
Memória de arquivos de usuário	16 Mbytes	16 Mbytes	32 Mbytes	32 Mbytes
Número máximo de tarefas	16	16	24	32
Número máximo de barramentos de expansão	1	8	24	24

Número máximo total de módulos de E/S nos barramentos	32	128	128	128
Ethernet TCP/IP interface local	1	1	2	2
Número máximo de módulos adicionais de interface Ethernet TCP/IP	0	0	2	6
Suporte a redundância de interfaces Ethernet TCP/IP	Não	Não	Sim	Sim
Número máximo de redes PROFIBUS-DP (usando módulos mestres PROFIBUS-DP)	1	1	4	4
Suporte a redundância de rede PROFIBUS-DP	Não	Não	Sim	Sim
Suporte a redundância (half-clusters)	Não	Não	Não	Sim
Registro de eventos (SOE)	Não	Não	Sim	Sim
Protocolo	-	-	DNP3	DNP3
Tamanho máximo da fila de eventos	-	-	1000	1000
Sincronização do relógio (SNTP)	Sim	Sim	Sim	Sim
Fonte de alimentação integrada	Sim	Não	Não	Não
Consumo de corrente no barramento da fonte de alimentação	-	800 mA	1000 mA	1000 mA
Dissipação	4 W	4 W	5 W	5 W

Tabela 2-5. Características Específicas

Notas:

Memória de variáveis de entrada de representação direta (%I): Área onde são alocadas as variáveis de representação direta para o tipo entrada. Variável de representação direta significa que a variável pode ser acessada diretamente na memória utilizando o endereço desejado. Por exemplo: %IB0, %IW100. Variável de entrada de representação direta pode ser utilizada para mapear pontos de entrada analógicos ou digitais. Como referência, 8 pontos de entrada digital podem ser representados por um byte e um ponto de entrada analógica pode ser representado por dois bytes.

Memória de variáveis de saída de representação direta (%Q) total: Área onde são alocadas todas as variáveis de representação direta para o tipo saída. Variável de representação direta significa que a variável pode ser acessada diretamente na memória utilizando o endereço desejado. Por exemplo: %QB0, %QW100. Variável de saída de representação direta pode ser utilizada para mapear pontos de saída analógicos ou digitais. Como referência, 8 pontos de saída digital podem ser representados por um byte e um ponto de saída analógica pode ser representado por dois bytes.

As variáveis de representação direta de saída ainda podem ser configuradas como retentivas, persistentes e/ou redundantes, mas o tamanho total não é alterado em função da configuração.

A UCP NX3030 da Série Nexto permite a definição de uma área de variáveis redundantes inseridas dentro da área de memória de variáveis de saída de representação direta %Q. O subconjunto de tipos de memória de variáveis de representação direta de saída fazem parte do total da memória disponibilizada.

Memória de variáveis de representação direta (%M): Área onde são alocadas as variáveis de representação direta para o tipo marker. Variável de representação direta significa que a variável pode ser acessada diretamente na memória utilizando o endereço desejado. Por exemplo: %MB0, %MW100.

Memória de variáveis simbólicas: Área onde são alocadas as variáveis simbólicas. As variáveis simbólicas são variáveis IEC criadas em POUs e GVLs durante o desenvolvimento do aplicativo, as quais não são endereçadas diretamente na memória. Variáveis simbólicas podem ser definida como retentivas ou persistentes, neste caso serão utilizadas as áreas de memória de variáveis simbólicas retentiva ou memória de variáveis simbólicas persistente respectivamente. O sistema do CP aloca variáveis nesta área, desta forma o espaço disponível para a alocação de variáveis criadas pelo usuário é inferior ao informado na tabela. A ocupação pelas variáveis de sistema depende das

características do projeto (quantidade de módulos, de drivers, etc...), desta forma recomenda-se observar o espaço disponível nas mensagens de compilação da ferramenta MasterTool IEC XE.

Memória de variáveis retentivas e persistentes: Área onde são alocadas as variáveis retentivas e/ou persistentes. Os dados retentivos mantêm seus respectivos valores mesmo após um ciclo de desenergização e energização da UCP. Já os dados persistentes mantêm seus respectivos valores mesmo após o download de uma nova aplicação na UCP.

ATENÇÃO:
A declaração e utilização de variáveis simbólicas persistentes devem ser realizadas única e exclusivamente através do objeto *Persistent Vars*, o qual pode ser incluído no projeto através da treeview em *Application -> Add Object -> Persistent Variables*. Não deve ser utilizada a expressão VAR PERSISTENT no campo de declaração de variáveis das POUs.

A lista completa de quando as variáveis retentivas e persistentes mantêm seus valores e quando o valor é perdido pode ser encontrada na Tabela 2-6. Além do tamanho de área persistente declarado na Tabela 2-5, estão reservados estes 44 bytes para armazenar informações sobre as variáveis persistentes (não disponível para uso).

A Tabela 2-6 mostra o comportamento das variáveis retentivas e persistentes para diferentes situações, em que "-" significa que o valor é perdido e "X" significa que o valor é mantido.

Comando	Variável simbólica	Variável retentiva	Variável persistente
Reset Quente/Ciclo de energização	-	X	X
Reset Frio	-	-	X
Reset Origem	-	-	-
Remover UCP ou Fonte de Alimentação do Bastidor enquanto energizado	-	-	-
Download	-	-	X
Alteração online	X	X	X
Reiniciar CP	-	X	X
Clean All	-	-	X

Tabela 2-6. Comportamento das Variáveis após Evento

Nas versões inferiores ou iguais a 1.5.0.21 para NX3004 e 1.5.1.0 para NX3010, NX3020 e NX3030 as memórias retentivas e persistentes simbólicas e de saída de representação direta (%Q) possuíam tamanho máximo fixo. Na Tabela 2-7 é possível consultar os tamanhos máximos permitidos nas versões antigas das UCPs.

Em versões superiores, as UCPs contam com a funcionalidade de tamanho de memórias retentivas e persistentes flexíveis. Para mais informações do funcionamento consulte a seção

Áreas de Memória Retentiva e Persistente em Configuração.

	NX3004	NX3010	NX3020	NX3030
Memória de variáveis de saída de representação direta retentivas (%Q)	2 Kbytes	8 Kbytes	16 Kbytes	16 Kbytes
Memória de variáveis de saída de representação direta persistentes (%Q)	2 Kbytes	8 Kbytes	16 Kbytes	48 Kbytes
Memória de variáveis simbólicas retentivas	2 Kbytes	32 Kbytes	48 Kbytes	32 Kbytes
Memória de variáveis simbólicas persistentes	1,5 Kbytes	16 Kbytes	32 Kbytes	16 Kbytes

Tabela 2-7. Memórias Retentivas e Persistentes em Versões Antigas

No caso do comando de Clean All, caso a aplicação tenha sido modificada de tal forma que variáveis persistentes tenham sido removidas, inseridas no início da lista ou então tenham tido o seu tipo modificado, o valor destas variáveis será perdido (alertado pela ferramenta MasterTool ao realizar o download). Desta forma recomenda-se que alterações na GVL de variáveis persistentes envolvam somente a inclusão de novas variáveis no final da lista.

Memória de dados redundantes total: Memória de dados redundantes é a quantidade máxima de memória que pode ser utilizada como memória redundante entre duas UCPs que formam o par redundante. Esse valor não se trata de uma memória diferente. Note que a soma de todas as variáveis redundantes (Variáveis de entrada de representação direta, Variáveis de saída de representação direta, Variáveis de representação direta, Variáveis simbólicas, Variáveis simbólicas retentivas, Variáveis simbólicas persistentes) deve ser igual ou menor que a memória de dados redundantes.

Memória de programa: Área da memória que corresponde ao tamanho máximo permitido para a aplicação de usuário. Essa área é compartilhada com a memória de código fonte, sendo a área total a soma de (Memória de programa + Memória de código fonte).

Memória de código fonte (backup): Área da memória utilizada como backup do projeto, ou seja, caso o usuário deseje importar o seu projeto, o software MasterTool IEC XE irá buscar as informações necessárias nessa área. É importante salientar que o usuário deve estar atento para não esquecer de atualizar o projeto que está salvo como backup toda vez que enviar a aplicação, evitando que informações sejam perdidas. Essa área é compartilhada com a memória de programa sendo a área total a soma de Memória de programa + Memória de código fonte.

Memória de arquivos de usuário: Essa área da memória é destinada ao armazenamento de arquivos, como: doc, pdf, imagens, entre outros; ou seja, permite a gravação de dados como se fosse um cartão de memória. Maiores informações no capítulo Configuração –

Memória de Arquivos de Usuário.

Número máximo de tarefas: O número máximo de tarefas definido para cada modelo de UCP e entre diferentes perfis de projeto, está melhor detalhado no capítulo Programação Inicial – Número Máximo de Tarefas.

Suporte a redundância (half-clusters): A versão de software 1.1.0.0 ou superior/revisão de produto AB ou superior suportam redundância de UCPs NX3030.

Registro de eventos (SOE): Os tipos de dados encontram-se no Anexo A.

Número máximo de módulos de E/S no barramento: O número máximo de módulos de E/S refere-se a soma de todos os módulos do barramento local e das expansões.

Número máximo de redes PROFIBUS-DP: Desde a versão 1.22 do MasterTool IEC XE ou superior é permitido 4 redes PROFIBUS-DP para as UCPs NX3020 e NX3030. Nas versões anteriores eram permitidas apenas 2 redes PROFIBUS-DP É limitado o uso de no máximo 4 módulos mestre PROFIBUS-DP, isto significa que apenas 2 redes redundantes podem ser utilizadas.

Interfaces Seriais

COM 1 (NX3010/NX3020/NX3030)

	NX3010, NX3020, NX3030
Conector	DB9 fêmea blindado
Meio físico	RS-232C
Sinais de modem	RTS, CTS, DCD
Baud rate	200, 300, 600, 1200, 1800, 2400, 4800, 9600, 19200, 38400, 57600, 115200 bps
Protocolos	MODBUS RTU mestre/escravo Protocolo aberto

Tabela 2-8. Características da Interface Serial COM 1

COM 1 (NX3004) e COM 2 (NX3010/NX3020/NX3030)

	NX3004, NX3010, NX3020, NX3030
Conector	DB9 fêmea blindado
Meio físico	RS-422 ou RS-485 (dependendo do cabo selecionado)
Direção de comunicação	RS-422: full duplex RS-485: half duplex
Número máximo de transceivers RS-422	11 (1 transmissor e 10 receptores)
Número máximo de transceivers RS-485	32
Terminação	Sim (opcional via seleção de cabo)
Baud rate	200, 300, 600, 1200, 1800, 2400, 4800, 9600, 19200, 38400, 57600, 115200 bps
Protocolos	MODBUS RTU mestre/escravo Protocolo aberto

Tabela 2-9. Características da Interface Serial COM 2

Notas:

Meio físico: Dependendo da configuração do cabo utilizado, é possível escolher o tipo de interface física: RS-422 ou RS-485. A lista dos cabos pode ser encontrada na seção Produtos Relacionados.

Número máximo de transceivers RS-422: O número máximo de transceivers é a maior quantidade de interfaces que podem ser utilizadas no mesmo barramento.

Número máximo de transceivers RS-485: O número máximo de transceivers é a maior quantidade de interfaces que podem ser utilizadas no mesmo barramento.

Interfaces Ethernet

NET 1

	NX3004, NX3010, NX3020, NX3030
Conector	RJ45 fêmea blindado
Auto negociação (auto crossover)	Sim
Comprimento máximo do cabo	100 m
Tipo de cabo	UTP ou ScTP, categoria 5
Baud rate	10/100 Mbps
Camada física	10/100 BASE-TX
Camada de enlace de dados	LLC (Controle de Enlace Lógico)
Camada de rede	IP (Protocolo de Internet)
Camada de transporte	TCP (Protocolo de Controle de Transmissão) e UDP (Protocolo de Datagrama de Usuário)
Camada de aplicação	Cliente/ Servidor MODBUS TCP Cliente/ Servidor MODBUS RTU via TCP HTTP (servidor web) Protocolo de programação MasterTool IEC XE DNP3 (relatório de dados orientados ao evento) SNTP (sincronismo de relógio) EtherCAT OPC SNMP (gerenciamento de rede Ethernet)
Diagnóstico	LEDs - verde (velocidade), amarelo (link/atividade)

Tabela 2-10. Características da Interface Ethernet NET 1

Nota:

Camada de aplicação: Os protocolos DNP3 e EtherCAT não estão disponíveis nas UCPs NX3004 e NX3010.

NET 2

	NX3020, NX3030
Conector	RJ45 fêmea blindado
Auto negociação (auto crossover)	Sim
Comprimento máximo do cabo	100 m
Tipo de cabo	UTP ou ScTP, categoria 5
Baud rate	10/100 Mbps
Camada física	10/100 BASE-TX
Camada de enlace de dados	LLC (Controle de Enlace Lógico)
Camada de rede	IP (Protocolo de Internet)
Camada de transporte	TCP (Protocolo de Controle de Transmissão) e UDP (Protocolo de Datagrama de Usuário)
Camada de aplicação	Cliente/ Servidor MODBUS TCP Cliente/ Servidor MODBUS RTU via TCP HTTP (servidor Web) Protocolo de programação MasterTool IEC XE DNP3 (relatório de dados orientados ao evento) SNTP (sincronismo de relógio) EtherCAT OPC SNMP (gerenciamento de rede Ethernet)
Diagnóstico	LEDs - verde (velocidade), amarelo (link/atividade)

Tabela 2-11. Características da Interface Ethernet NET 2

Fonte de Alimentação

	NX3004
Tensão de entrada nominal	24 Vdc
Potência de saída máxima	15 W
Corrente de saída máxima	3 A
Tensão de entrada	19,2 a 30 Vdc
Corrente de entrada máxima (in-rush)	30 A
Corrente de entrada máxima	1,4 A
Eficiência típica	92 % @ 24 Vdc
Tempo máximo de interrupção da tensão de entrada	10 ms
Isolação	
Entrada para saída	1000 Vac / 1 minuto
Entrada para terra de proteção	1500 Vac / 1 minuto
Entrada para terra funcional	1000 Vac / 1 minuto
Bitola do fio	0,5 mm²
Proteção inversão de polaridade	Sim
Fusível rearmável interno	Sim
Proteção contra curto-circuito na saída	Sim
Proteção contra sobrecorrente	Sim

Tabela 2-12. Características da Fonte de Alimentação

Interface do Cartão de Memória

Os cartões de memória podem ser usados para diferentes tipos de armazenamento de dados como: logs de usuários, páginas web, documentação de projeto e arquivos fontes. Mais informações sobre como usar interface de cartão de memória podem ser encontradas em Cartão de Memória.

	NX3010, NX3020, NX3030
Capacidade máxima	8 Gbytes
Capacidade mínima	2 Gbytes
Tipo	miniSD
Sistema de arquivos	FAT32
Remover cartão de forma segura	Sim, pressionando o botão MS

Tabela 2-13. Características da Interface com Cartão de Memória

Notas:

Capacidade máxima: A capacidade do cartão de memória deve ser igual ou inferior a este limite para o correto funcionamento na UCP Nexto, podendo a UCP não reconhecer o cartão ou ocorrer perdas de dados durante transferências.

Capacidade mínima: A capacidade do cartão de memória deve ser igual ou superior a este limite para o seu correto funcionamento na UCP Nexto, podendo a UCP não reconhecer o cartão ou ocorrer perdas de dados durante transferências.

Sistema de arquivos: É recomendado formatar a memória utilizando a própria UCP Nexto, caso contrário poderá ocorrer perda de desempenho no acesso a interface do cartão de memória.

Compatibilidade com Demais Produtos

Existem algumas incompatibilidades entre às UCPs da Série Nexto e o MasterTool IEC XE. Consulte a tabela a seguir para saber qual versão de MasterTool IEC XE deve ser utilizada.

Versão de software das UCPs da Série Nexto		Versão de MasterTool IEC XE compatível
NX3004	NX3010, NX3020, NX3030	
-	**1.2.0.9 ou inferior**	1.00 a 1.28
-	**1.2.1.0 a 1.2.4.0**	1.29 a 1.40
-	**1.3.0.20**	1.40 a 1.41
-	**1.4.0.33**	2.00
1.5.0.18	**1.5.0.10 a 1.5.0.16**	2.01 a 2.02
1.5.0.21	**1.5.1.0**	2.03 ou superior

Tabela 2-14. Compatibilidade entre produtos

Nota:

Compatibilidade entre versões: Existem algumas limitações quanto à funcionalidades, algumas estão disponíveis apenas a partir de determinada versão. Consulte a seção do Manual de Utilização MasterTool IEC XE – MU299048, sobre a funcionalidade para verificar a disponibilidade de determinada característica em uma versão específica do programador.

Desempenho

O desempenho das UCPs da Série Nexto depende:

- Tempo da Aplicação do Usuário
- Intervalo da Aplicação
- Tempo do Sistema Operacional
- Quantidade de módulos (dados de processo, entradas/saídas, entre outros)

Tempos de Aplicação

O tempo de execução de uma aplicação das UCPs da Série Nexto depende das seguintes variáveis:

- Tempo de leitura das entradas (locais e remotas)
- Tempo de execução das tarefas
- Tempo de escrita das saídas (locais e remotas)

É importante ressaltar que o tempo de execução da tarefa "MainTask" será diretamente influenciado pela tarefa de sistema "Configuration", uma tarefa de alta prioridade, executada periodicamente pelo sistema. A tarefa "Configuration" poderá interromper a "MainTask" e, quando utilizados módulos de comunicação, como, por exemplo, o módulo Ethernet NX5000, o acréscimo de tempo à "MainTask" poderá ser de até 25% do seu tempo médio de execução.

Tempos para Execução de Instruções

Na Tabela 2-15, encontram-se os tempos necessários para a realização de diferentes instruções nas UCPs da Série Nexto:

Instrução	Linguagem	Variáveis	Tempos de Execução (us)
1000 Contatos	LD	BOOL	6
1000 Divisões	ST	INT	43
		REAL	81
	IL	INT	43
		REAL	81
	LD	INT	43
		REAL	81
1000 Multiplicações	ST	INT	15
		REAL	23
	IL	INT	15
		REAL	23
	LD	INT	15
		REAL	23
1000 Somas	ST	INT	15
		REAL	23
	IL	INT	15
		REAL	23
	LD	INT	15
		REAL	23

Tabela 2-15. Tempos de Instruções

Tempos de Inicialização

As UCPs da Série Nexto possuem tempo de inicialização de 50 segundos, sendo que a tela inicial com o logotipo é apresentada depois de 20 segundos da energização.

Tempo de Intervalo

O tempo de Intervalo de cada tarefa da UCP é dependente do aplicativo, podendo ser configurado com valores de 5 a 750 ms.

Dimensões Físicas

NX3004

Dimensões em mm.

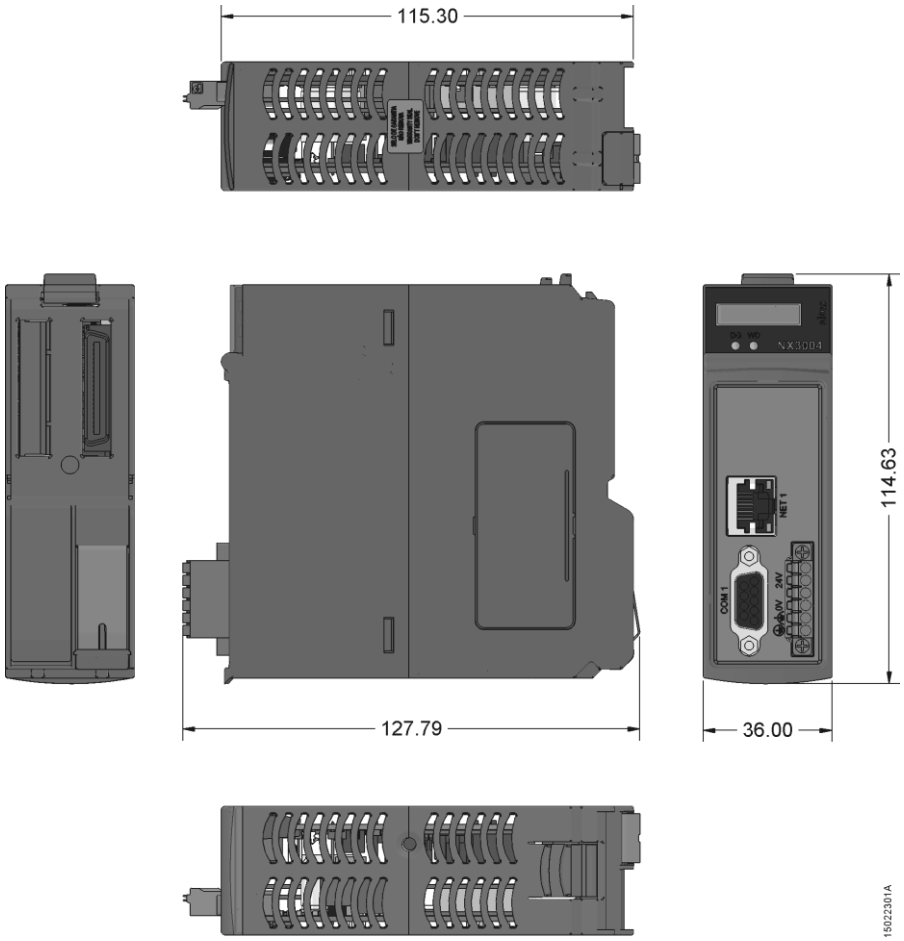

Figura 2-2. Dimensões Físicas da UCP NX3004

NX3010, NX3020 e NX3030

Dimensões em mm.

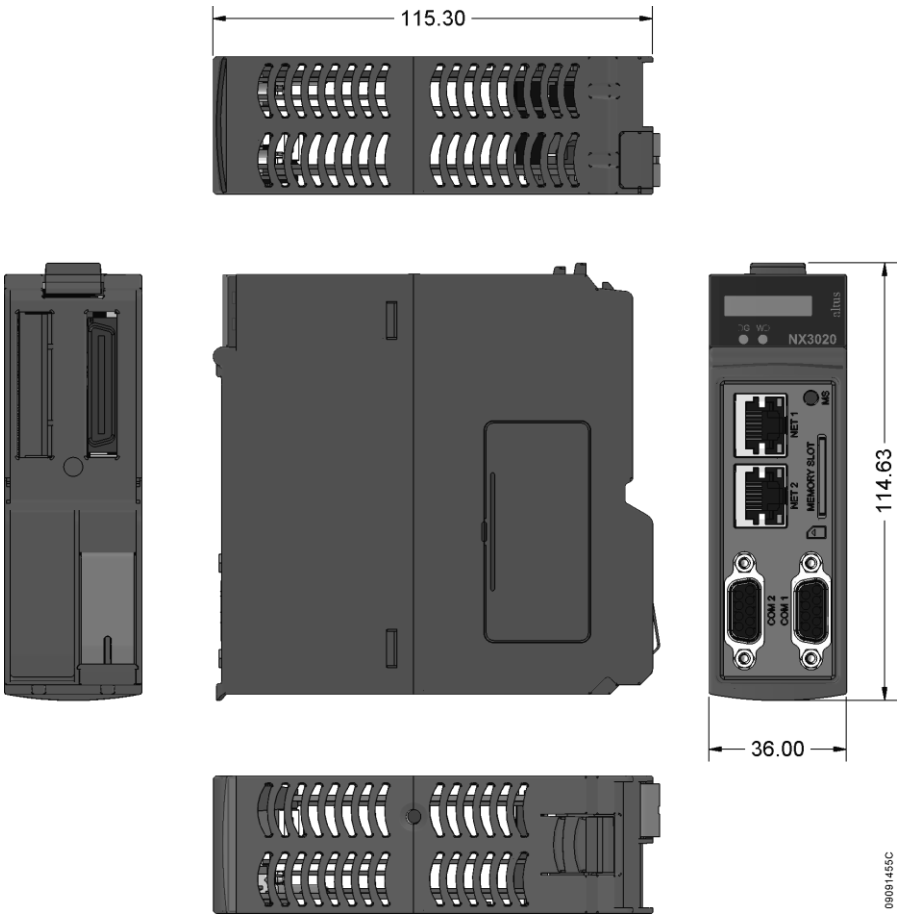

Figura 2-3. Dimensões Físicas das UCPs NX3010, NX3020 e NX3030

Dados para Compra

Itens Integrantes

A embalagem do produto contém os seguintes itens:

- Módulo NX3004, NX3010, NX3020 ou NX3030
- Conector 6 terminais com fixação (somente para NX3004)
- Guia de Instalação

Código do Produto

Os seguintes códigos devem ser usados para compra do produto:

Código	Descrição
NX3004	UCP com 1 porta Ethernet, 1 canal serial, suporte a expansão de barramento e fonte de alimentação integrada
NX3010	UCP de alta velocidade, 1 porta Ethernet, 2 canais seriais, interface para cartão de memória e suporte a expansão de barramento
NX3020	UCP de alta velocidade, 2 portas Ethernet, 2 canais seriais, interface para cartão de memória e suporte a expansão de barramento
NX3030	UCP de alta velocidade, 2 portas Ethernet, 2 canais seriais, interface para cartão de memória, suporte a expansão de barramento e suporte a redundância

Tabela 2-16. Modelos de UCPs da Série Nexto

Produtos Relacionados

Os seguintes produtos devem ser adquiridos separadamente quando necessário:

Código	Descrição
MT8500	MasterTool IEC XE
AL-2600	Derivador e terminador de rede RS-485
AL-2301	Cabo de rede RS-485 (até 1000 metros)
AL-2306	Cabo de rede RS-485 (até 500 metros)
AL-2319	Cabo RJ45-RJ45
AL-1729	Cabo RJ45-CMDB9
AL-1748	Cabo CMDB9-CFDB9
AL-1752	Cabo CMDB9-CMDB9
AL-1753	Cabo CMDB9-CMDB25
AL-1754	Cabo CMDB9-CFDB9
AL-1761	Cabo CMDB9- CMDB9
AL-1762	Cabo CMDB9- CMDB9
AL-1763	Cabo CMDB9-Terminais
NX9202	Cabo RJ45-RJ45 2 m
NX9205	Cabo RJ45-RJ45 5 m
NX9210	Cabo RJ45-RJ45 10 m

Tabela 2-17. Produtos Relacionados

Notas:

MT8500: MasterTool IEC XE está disponível em quatro diferentes versões: LITE, BASIC, PROFESSIONAL e ADVANCED. Para maiores informações, favor consultar o Manual de Utilização do MasterTool IEC XE - MU299048.

AL-2600: Este módulo é utilizado para derivação e terminação de uma rede RS-422/485. Para cada nó da rede deve existir um AL-2600. Os módulos AL-2600 que estiverem nas extremidades da rede

devem ser configurados como terminação, exceto quando há um dispositivo com terminação interna ativa, sendo o restante como derivação.

AL-2301: Cabo blindado de dois pares trançados, sem conectores, para ser utilizado em redes RS-485 e redes RS-422, com comprimento máximo de 1000 metros.

AL-2306: Cabo blindado de dois pares trançados, sem conectores, para ser utilizado em redes RS-485 e redes RS-422 com comprimento máximo de 500 metros.

AL-2319: Cabo com dois conectores RJ45 para programação das UCPs da Série Nexto e para comunicação Ethernet ponto-a-ponto com outro dispositivo com interface Ethernet.

AL-1729: Cabo padrão RS-232C com um conector RJ45 e um conector DB9 macho para comunicação entre as UCPs da Série Nexto e outros produtos Altus das Séries DUO, Piccolo e Ponto.

AL-1748: Cabo padrão RS-232C com um conector DB9 macho e um conector DB9 fêmea para comunicação entre as UCPs da Série Nexto e outros produtos Altus da Série Cimrex.

AL-1752: Cabo padrão RS-232C com dois conectores DB9 macho para comunicação entre as UCPs da Série Nexto e outros produtos Altus da Série H e Série iX.

AL-1753: Cabo padrão RS-232C com um conector DB9 macho e um conector DB25 macho para comunicação entre as UCPs da Série Nexto e outros produtos Altus da Série H.

AL-1754: Cabo padrão RS-232C com um conector DB9 macho e um conector DB9 fêmea para comunicação entre as UCPs da Série Nexto e outros produtos Altus da Série Exter ou porta serial padrão RS-232C de um microcomputador.

AL-1761: Cabo padrão RS-232C com dois conectores DB9 macho para comunicação entre as UCPs da Série Nexto e outros produtos Altus da Série AL.

AL-1762: Cabo padrão RS-232C com dois conectores DB9 macho para comunicação entre as UCPs da Série Nexto.

AL-1763: Cabo com um conector DB9 macho e terminais para comunicação entre as UCPs da Série Nexto e produtos com bornes padrão RS-485/RS-422.

NX9202/NX9205/NX9210: Cabos utilizados para interligar módulos expansores de barramento.

3. Instalação

Este capítulo apresenta os procedimentos necessários para a instalação física das UCPs da Série Nexto, bem como os cuidados que se deve ter com outras instalações existentes no armário elétrico ocupado pelo CP.

Instalação Mecânica

As UCPs da Série Nexto (NX3010, NX3020 e NX3030) devem ser inseridas na posição 2 do bastidor, logo após o Módulo Fonte de Alimentação. Todas as informações sobre a montagem mecânica e inserção do módulo podem ser encontradas no Manual de Utilização Série Nexto - MU214000.

Instalação Elétrica

PERIGO:
Ao realizar qualquer instalação em um painel elétrico, certifique-se de que a alimentação geral do armário esteja DESLIGADA.

NX3004

A UCP NX3004 deve estar posicionada na posição 0 do bastidor da Série Nexto. São necessárias duas posições sequenciais, isto significa que esta ocupará as posições 0 e 1 de um dado bastidor.

A Figura 3-1 ilustra o diagrama elétrico da UCP NX3004 instalada em um bastidor da Série Nexto.

A disposição dos conectores e bornes na figura é meramente ilustrativa.

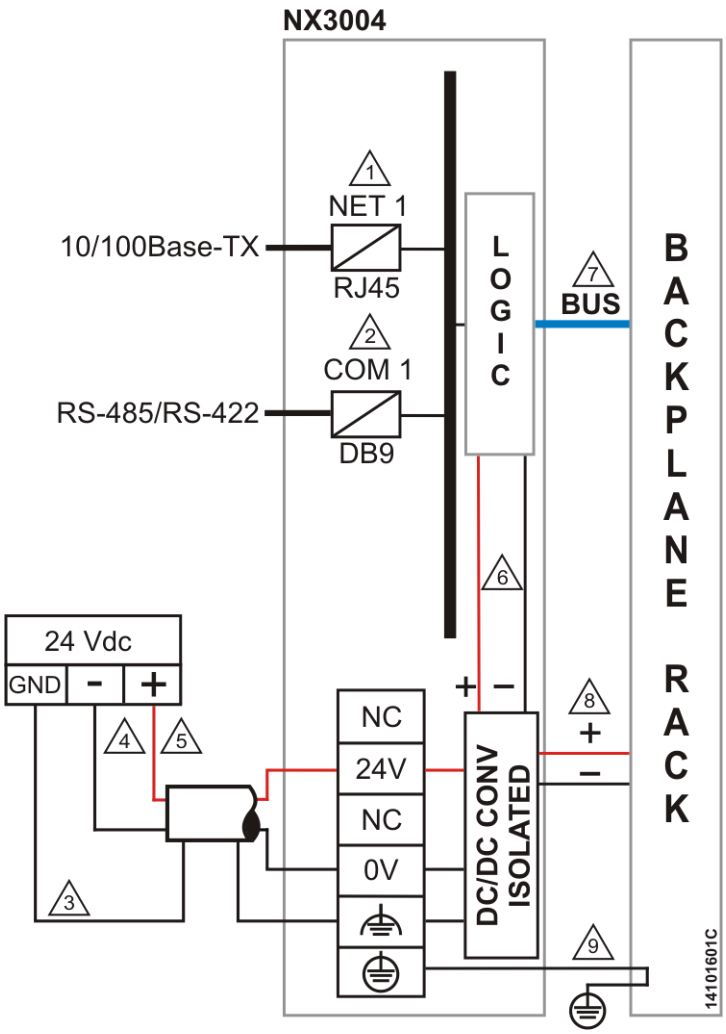

Figura 3-1. Diagrama Elétrico da UCP NX3004 da Série Nexto

Notas do Diagrama:

1. Interface Ethernet padrão 10/100Base-TX para programação, depuração e conexão à rede MODBUS TCP ou outros protocolos.
2. Interface serial padrão RS-485/RS-422 para conexão à rede MODBUS RTU ou outros protocolos. A escolha do tipo de interface física depende do cabo utilizado.
3. O aterramento vindo da fonte de alimentação externa está conectado ao terminal ⏚. Utilizar cabos de 0,5 mm².
4. A fonte de alimentação está conectada ao terminal 0 V. Utilizar cabos de 0,5 mm².
5. A fonte de alimentação está conectada ao terminal 24 V. Utilizar cabos de 0,5 mm².
6. A fonte de alimentação alimenta o circuito interno diretamente.
7. Barramento local de dados.
8. O módulo alimenta os outros módulos da Série Nexto através da conexão com o bastidor.
9. O aterramento do módulo é feito através do bastidor da Série Nexto.

NX3010, NX3020 e NX3030

A alimentação das UCPs NX3010, NX3020 e NX3030 é proveniente do Módulo Fonte de Alimentação, o qual fornece tensão às UCPs através da conexão ao bastidor, não necessitando de conexões externas. O aterramento do módulo é realizado através do contato entre a mola de aterramento do módulo e o bastidor.

A Figura 3-2 ilustra o diagrama elétrico das UCPs da Série Nexto instalada em um bastidor da Série Nexto.

A disposição dos conectores e bornes na figura é meramente ilustrativa.

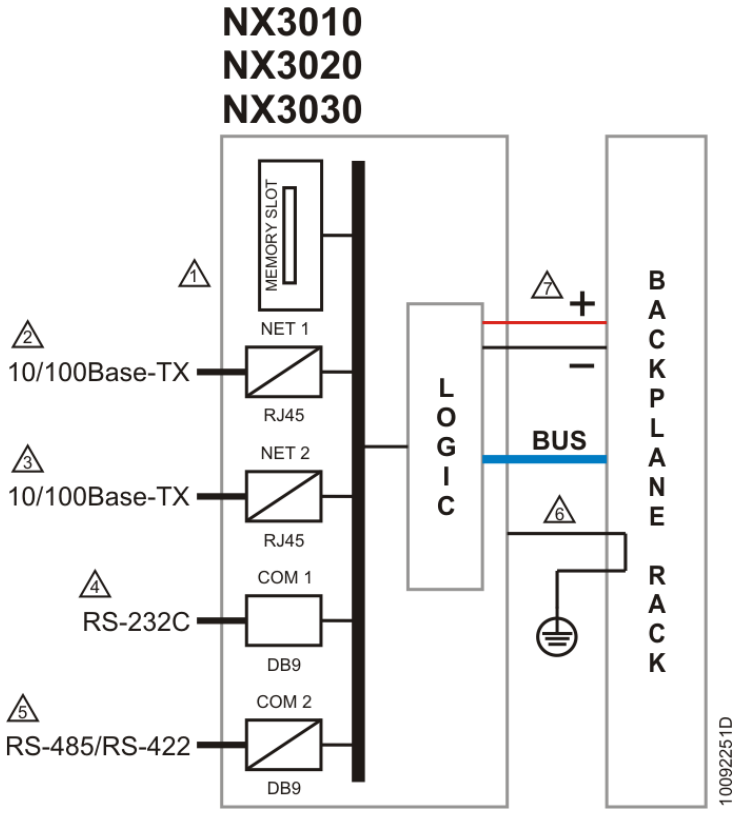

Figura 3-2. Diagrama Elétrico das UCPs NX3010, NX3020 e NX3030 da Série Nexto

Notas do Diagrama:

1. Interface para cartão de memória.
2. Interface Ethernet padrão 10/100Base-TX para programação, depuração e conexão à rede MODBUS TCP ou outros protocolos.
3. Interface Ethernet padrão 10/100Base-TX para conexão à rede MODBUS TCP ou outros protocolos (somente para NX3020 e NX3030).
4. Interface serial padrão RS-232C para conexão à rede MODBUS RTU ou outros protocolos.
5. Interface serial padrão RS-485/RS-422 para conexão à rede MODBUS RTU ou outros protocolos. A escolha do tipo de interface física depende do cabo utilizado.
6. O módulo é aterrado através dos bastidores da Série Nexto.
7. A alimentação do módulo é proveniente da conexão ao bastidor, não necessitando de conexões externas.

Conexão com a Rede Ethernet

As interfaces isoladas de comunicação NET 1 e NET 2 (somente para NX3020 e NX3030) possibilitam a conexão com uma rede Ethernet, onde a interface NET 1 é mais indicada para comunicação com o MasterTool IEC XE.

A conexão com a rede Ethernet utiliza cabos tipo par trançado (10/100Base-TX), sendo que a detecção da velocidade é realizada automaticamente pela UCP Nexto. Este cabo deve ter uma de suas extremidades ligadas à interface que se pretende utilizar e a outra ao HUB, switch, microcomputador ou outro ponto de rede Ethernet.

Endereço IP

A interface de Ethernet NET 1 é utilizada para comunicação Ethernet e para configurar a UCP, para que isso seja possível, esta vem configurada de fábrica com os seguintes parâmetros:

	NET 1
Endereço IP	192.168.15.1
Máscara de subrede	255.255.255.0
Endereço do gateway	192.168.15.253

Tabela 3-1. Parâmetros de Fábrica da Interface Ethernet NET 1

Os parâmetros Endereço IP e Máscara de Subrede podem ser visualizados no visor gráfico da UCP através do menu de parâmetros, conforme descrito no capítulo Configuração – Menu Informativo e de Configuração da UCP.

Inicialmente, deve-se conectar a interface NET 1 da UCP a uma rede ou microcomputador com a mesma subrede para comunicação com o MasterTool IEC XE, onde os parâmetros de rede podem ser alterados. Para maiores detalhes sobre configuração e alteração de parâmetros de rede, verifique o capítulo Configuração – Configuração das Interfaces Ethernet.

A interface de Ethernet NET 2 disponível somente para as UCPs NX3020 e NX3030 é utilizada para comunicação Ethernet e vem configurada de fábrica com os seguintes parâmetros:

	NET 2
Endereço IP	192.168.16.1
Máscara de subrede	255.255.255.0
Endereço do gateway	192.168.16.253

Tabela 3-2. Parâmetros de Fábrica da Interface Ethernet NET 2

Os parâmetros Endereço IP e Máscara de Subrede podem ser visualizados no visor gráfico da UCP através do menu de parâmetros, conforme descrito no capítulo Configuração –Menu Informativo e de Configuração da UCP.

Os parâmetros de rede da interface NET 2 podem ser alterados através do MasterTool IEC XE. Para maiores detalhes sobre configuração e alteração de parâmetros de rede, verifique o capítulo Configuração – Configuração das Interfaces Ethernet.

ARP Gratuito

A interface de Ethernet NETx envia espontaneamente pacotes do tipo ARP, em broadcast, informando seu endereço de IP e MAC para todos os dispositivos interligados à rede. Estes pacotes são enviados durante o download de uma nova aplicação pelo software MasterTool IEC XE e na inicialização da UCP, quando a aplicação entra em modo Run.

São disparados 5 comandos ARP com um intervalo inicial de 200 ms, dobrando o intervalo entre cada novo comando disparado, totalizando 3 s. Ex.: Primeiro disparo ocorre no tempo 0, o segundo disparo no tempo 200 ms, o terceiro disparo no tempo 600 ms e assim até o quinto disparo no tempo 3 s.

Instalação do Cabo de Rede

As portas Ethernet das UCPs da Série Nexto, identificadas no painel por NET 1 e NET 2 (NX3020 e NX3030), possuem pinagem padrão, sendo a mesma utilizada, por exemplo, em computadores pessoais. O tipo de conector, tipo de cabo, nível físico, entre outros detalhes para interligar a UCP ao dispositivo de acesso à rede Ethernet, são definidos no capítulo Descrição Técnica - Interfaces Ethernet.

A Figura 3-3 e a Tabela 3-3 apresentam o conector RJ45 fêmea das UCPs Nexto, com a identificação e a descrição da pinagem válida para os níveis físicos tipo 10Base-TX e 100Base-TX.

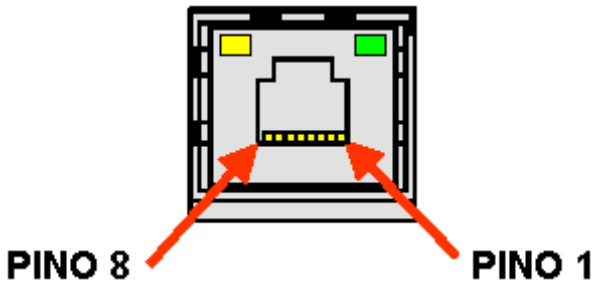

Figura 3-3. Conector RJ45 fêmea das UCPs Nexto

Pino	Sinal	Descrição
1	TXD+	Transmissão de dados, positivo
2	TXD-	Transmissão de dados, negativo
3	RXD+	Recepção de dados, positivo
4	-NU	Não utilizado
5	-NU	Não utilizado
6	RXD-	Recepção de dados, negativo
7	NU	Não utilizado
8	NU	Não utilizado

Tabela 3-3. Pinagem do Conector RJ45 Fêmea das UCPs Nexto

A interface pode ser conectada em uma rede de comunicação através de um hub ou switch, ou então diretamente ao equipamento com o qual irá se comunicar. Neste último caso, devido as UCPs Nexto possuírem a característica de Auto Crossover, não se faz necessária a utilização de um cabo de rede denominado cross-over, o mesmo utilizado para conectar dois computadores pessoais, ponto a ponto, através da porta Ethernet.

É importante ressaltar que entende-se por cabo de rede, um par de conectores RJ45 machos interligados entre si por um cabo UTP ou ScTP, de categoria 5, sob a configuração direta ou cross-over. O mesmo serve para interligar dois dispositivos com porta Ethernet.

Normalmente estes cabos possuem uma trava de conexão que garante uma perfeita conexão entre o conector fêmea da interface e o conector macho do cabo. No momento da instalação, o conector macho do cabo deve ser inserido na fêmea do módulo até que se ouça um som característico ("click"), garantindo a atuação da trava. Para desconectar os mesmos deve-se utilizar a alavanca presente no conector macho.

Conexão com a Rede Serial RS-232C

A interface não isolada de comunicação COM 1 das UCPs NX3010, NX3020 e NX3030 possibilita a conexão com uma rede RS-232C. A seguir é apresentado o conector DB9 fêmea da UCP Nexto, com a identificação e a descrição dos sinais.

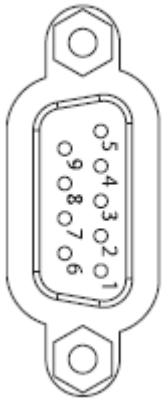

Figura 3-4. Conector DB9 Fêmea COM 1 das UCPs NX3010, NX3020 e NX3030

Pino	Sinal	Descrição
1	DCD	Data Carrier Detect
2	TXD	Transmissão de dados
3	RXD	Recepção de dados
4	-	Não utilizado
5	GND	Ground
6	-	Não utilizado
7	CTS	Clear to Send
8	RTS	Request to Send
9	-	Não utilizado

Tabela 3-4. Pinagem do Conector DB9 Fêmea COM 1 (NX3010, NX3020 e NX3030)

Comunicação RS-232C

Para conexão com um dispositivo RS-232C, utilizar o cabo apropriado conforme o capítulo Descrição Técnica – Produtos Relacionados.

Conexão com a Rede Serial RS-485/422

As interfaces isoladas de comunicação COM1, para a UCP NX3004, e COM 2 , para as UCPs NX3010, NX3020 ou NX3030, possibilitam a conexão com uma rede RS-485/422. A seguir é apresentado o conector DB9 fêmea da UCP Nexto, com a identificação e a descrição dos sinais.

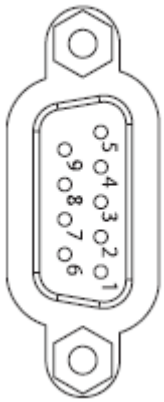

Figura 3-5. Conector DB9 Fêmea COM 1 (NX3004) ou COM 2 (NX3010, NX3020 e NX3030)

Pino	Sinal	Descrição
1	-	Não utilizado
2	Term+	Terminação interna, positivo
3	TXD+	Transmissão de dados, positivo
4	RXD+	Recepção de dados, positivo
5	GND	Referência negativa para terminação externa
6	+5V	Referência positiva para terminação externa
7	Term-	Terminação interna, negativo
8	TXD-	Transmissão de dados, negativo
9	RXD-	Recepção de dados, negativo

Tabela 3-5. Pinagem do Conector DB9 Fêmea COM 1 (NX3004) ou COM 2 (NX3010, NX3020 e NX3030)

Comunicação RS-485 sem Terminação

Para conexão a uma rede RS-485 sem terminação na interface COM 1 (NX3004) ou COM 2 (NX3010, NX3020 ou NX3030), deve-se conectar os terminais identificados do cabo AL-1763 nos respectivos bornes do dispositivo, conforme a Tabela 3-6.

Terminais do AL-1763	Sinais na Borneira da UCP
0	Blindagem
1	Não conectado
2	D+
3	D+
4	Não conectado
5	Não conectado
6	Não conectado
7	D-
8	D-

Tabela 3-6. Conexão da Interface RS-485 sem Terminação

O diagrama da Figura 3-6 indica como deve ser realizada a conexão dos terminais do AL-1763 na borneira do dispositivo.

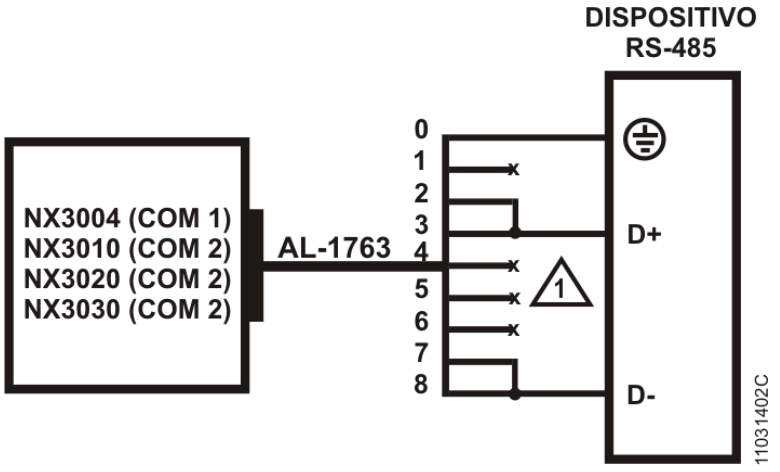

Figura 3-6. Diagrama de Conexão da Interface RS-485 sem Terminação

Nota do Diagrama:

Os terminais não conectados devem ser isolados para que não haja contato entre os mesmos.

Comunicação RS-485 com Terminação Interna

Para conexão a uma rede RS-485, utilizando a terminação interna da interface COM 1 (NX3004) ou COM 2 (NX3010, NX3020 ou NX3030), deve-se conectar os terminais identificados do cabo AL-1763 nos respectivos bornes do dispositivo, conforme a Tabela 3-7.

Terminais do AL-1763	Sinais na Borneira da UCP
0	Blindagem
1	D+
2	D+
3	D+
4	Não conectado
5	Não conectado
6	D-
7	D-
8	D-

Tabela 3-7. Conexão da Interface RS-485 com Terminação Interna

Obs.: As terminações internas disponíveis na COM 1 (NX3004) e na COM 2 (NX3010, NX3020 ou NX3030) são do tipo estado seguro em modo aberto.

O diagrama da Figura 3-7 indica como deve ser realizada a conexão dos terminais do AL-1763 na borneira do dispositivo.

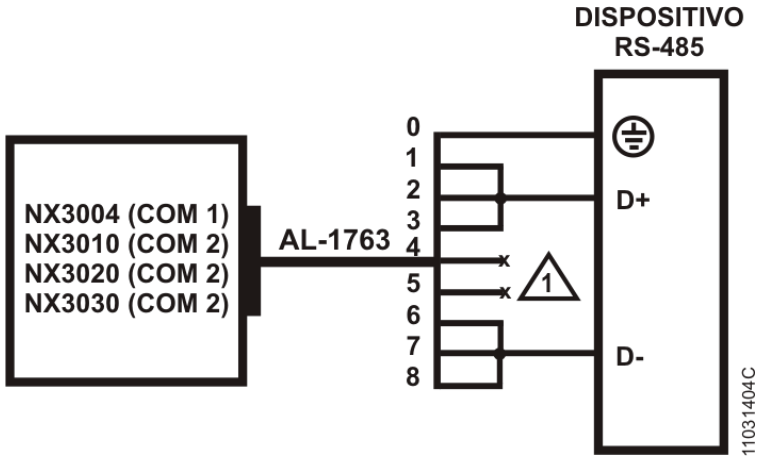

Figura 3-7. Diagrama de Conexão da Interface RS-485 com Terminação Interna

Nota do Diagrama:

Os terminais não conectados devem ser isolados para que não haja contato entre os mesmos.

Comunicação RS-485 com Terminação Externa

Para conexão a uma rede RS-485, utilizando a terminação externa da interface COM 1 (NX3004) ou COM 2 (NX3010, NX3020 ou NX3030), deve-se conectar os terminais identificados do cabo AL-1763 nos respectivos bornes do dispositivo, conforme a Tabela 3-8.

Terminais do AL-1763	Sinais na Borneira da UCP
0	Blindagem
1	Não conectado
2	D+
3	D+
4	0 V
5	+5 V
6	Não conectado
7	D-
8	D-

Tabela 3-8. Conexão da Interface RS-485 com Terminação Externa

O diagrama da Figura 3-8 indica como deve ser realizada a conexão dos terminais do AL-1763 na borneira do dispositivo.

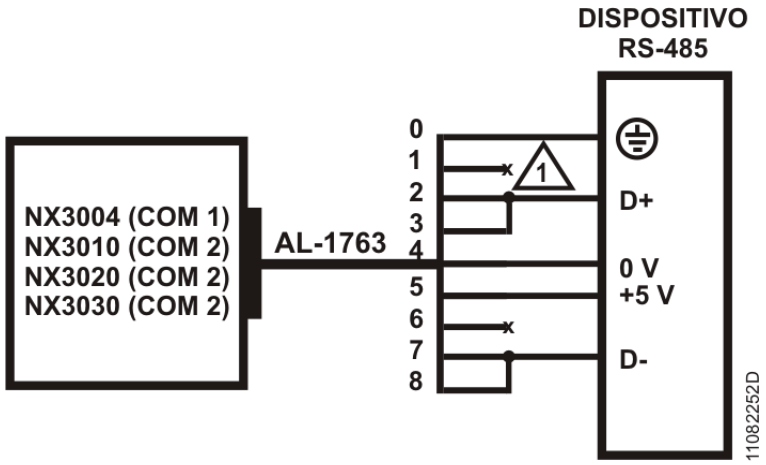

Figura 3-8. Diagrama de Conexão da Interface RS-485 com Terminação Externa

Nota do Diagrama:

Os terminais não conectados devem ser isolados para que não haja contato entre os mesmos.

Exemplo de Ligação de Rede RS-485 com Terminação Externa e Redundância de Mestre

Abaixo, a Figura 3-9 mostra um exemplo de ligação da rede RS-485 com terminação externa, utilizando duas UCPs Nexto NX3030 com redundância de half-cluster, como mestre.

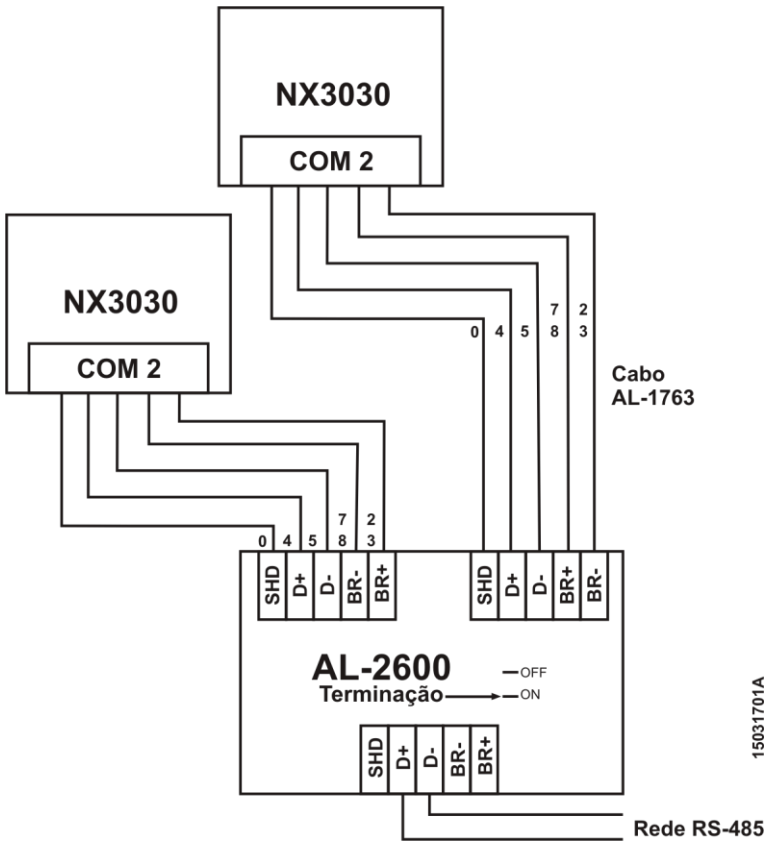

Figura 3-9. Diagrama de Conexão de Rede RS-485 com Terminação Externa e Redundância de Mestre

Comunicação RS-422 sem Terminação

Para conexão a uma rede RS-422 sem terminação na interface COM 1 (NX3004) ou COM 2 (NX3010, NX3020 ou NX3030), deve-se conectar os terminais identificados do cabo AL-1763 nos respectivos bornes do dispositivo, conforme a Tabela 3-9.

Terminais do AL-1763	Sinais na Borneira da UCP
0	Blindagem
1	Não conectado
2	TX+
3	RX+
4	Não conectado
5	Não conectado
6	Não conectado
7	TX-
8	RX-

Tabela 3-9. Conexão da Interface RS-422 sem Terminação

O diagrama da Figura 3-10 indica como deve ser realizada a conexão dos terminais do AL-1763 na borneira do dispositivo.

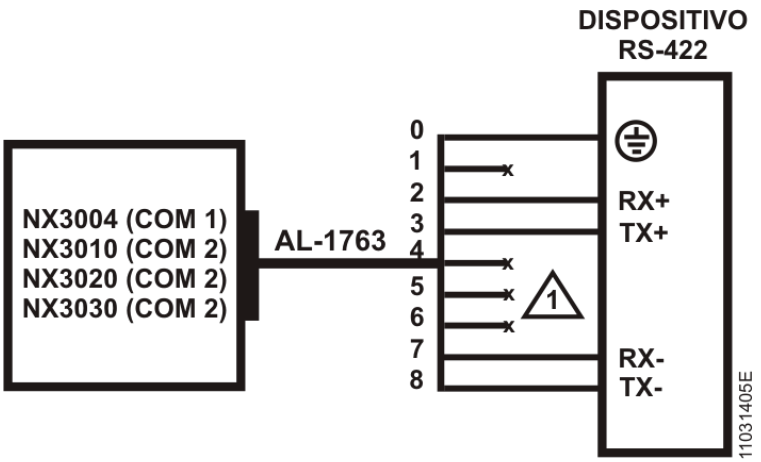

Figura 3-10. Diagrama de Conexão da Interface RS-422 sem Terminação

Nota do Diagrama:

Os terminais não conectados devem ser isolados para que não haja contato entre os mesmos.

Comunicação RS-422 com Terminação Interna

Para conexão a uma rede RS-422, utilizando a terminação interna da interface COM 1 (NX3004) ou COM 2 (NX3010, NX3020 ou NX3030), deve-se conectar os terminais identificados do cabo AL-1763 nos respectivos bornes do dispositivo, conforme a Tabela 3-10.

Terminais do AL-1763	Sinais na Borneira da UCP
0	Blindagem
1	TERM+
2	TX+
3	RX+
4	Não conectado
5	Não conectado
6	TERM-
7	TX-
8	RX-

Tabela 3-10. Conexão da Interface RS-422 com Terminação

Obs.: As terminações internas disponíveis na COM 1 (NX3004) e na COM 2 (NX3010, NX3020 ou NX3030) são do tipo estado seguro em modo aberto.

O diagrama da Figura 3-11 indica como deve ser realizada a conexão dos terminais do AL-1763 na borneira do dispositivo.

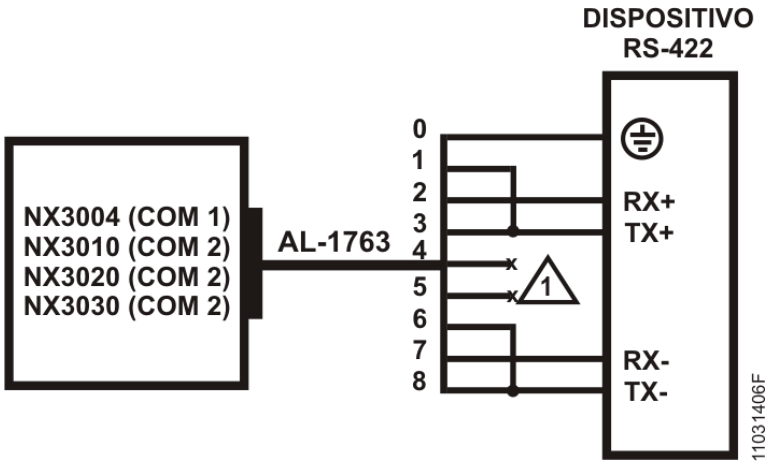

Figura 3-11. Diagrama de Conexão da InterfaceRS-422 com Terminação

Nota do Diagrama:

Os terminais não conectados devem ser isolados para que não haja contato entre os mesmos.

Comunicação RS-422 com Terminação Externa

Para conexão a uma rede RS-422, utilizando a terminação externa da interface COM 1 (NX3004) ou COM 2 (NX3010, NX3020 ou NX3030), deve-se conectar os terminais identificados do cabo AL-1763 nos respectivos bornes do dispositivo, conforme a Tabela 3-11.

Terminais do AL-1763	Sinais na Borneira da UCP
0	Blindagem
1	Não conectado
2	TX+
3	RX+
4	0 V
5	+5 V
6	Não Conectado
7	TX-
8	RX-

Tabela 3-11. Conexão da InterfaceRS-422 com Terminação Externa

O diagrama da Figura 3-12 indica como deve ser realizada a conexão dos terminais do AL-1763 na borneira do dispositivo.

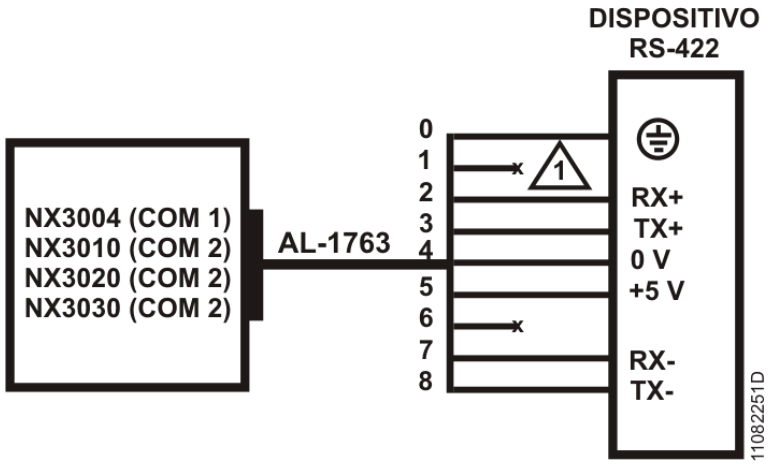

Figura 3-12. Diagrama de Conexão da InterfaceRS-422 com Terminação Externa

Nota do Diagrama:

Os terminais não conectados devem ser isolados para que não haja contato entre os mesmos.

Exemplo de Rede RS-422

Abaixo, a Figura 3-13 mostra um exemplo de utilização da rede RS-422, utilizando a UCP Nexto como mestre, dispositivos escravos com interface RS-422, e solução Altus para derivadores e terminadores.

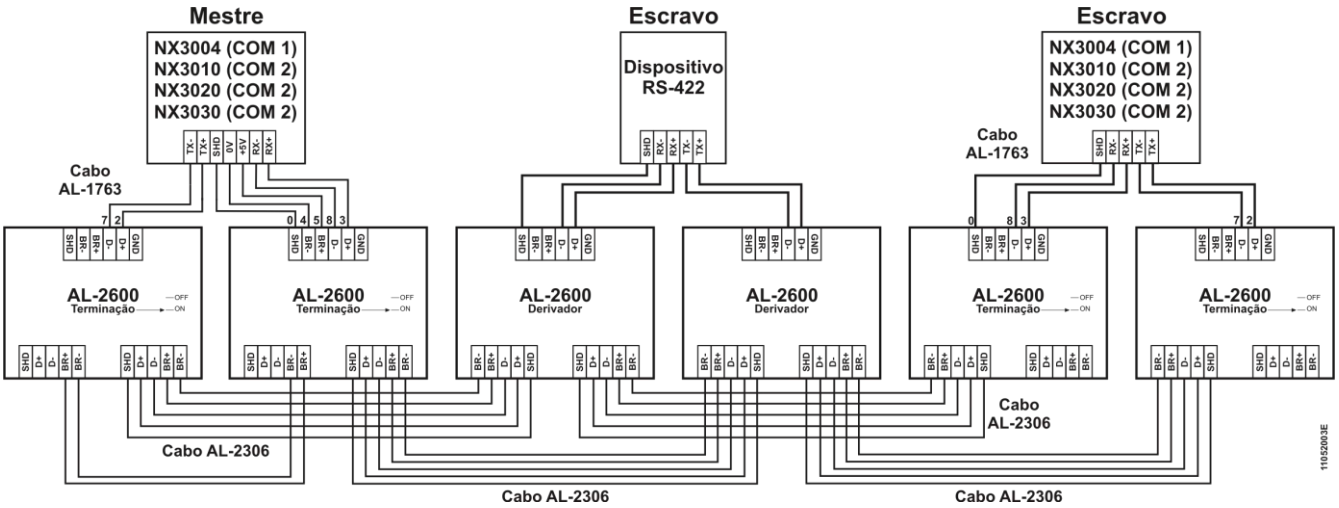

Figura 3-13. Exemplo de Rede RS-422

Nota do Diagrama:

Os módulos AL-2600 que estão nas extremidades da rede fazem a função de terminadores. Neste caso deve-se configura as chaves do AL-2600 em Terminação PROFIBUS.

Instalação do Cartão de Memória

Nesta seção está descrita a maneira como o cartão de memória deve ser inserido nas UCPs da Série Nexto modelos NX3010, NX3020 e NX3030, sendo que, para maiores informações sobre a utilização do mesmo, consultar o capítulo Configuração – Cartão de Memória.

Inicialmente, deve-se ter atenção quanto à posição correta que o cartão de memória deve ser inserido. Um dos cantos se diferencia dos demais e o mesmo deverá ser utilizado como referência para a inserção correta do cartão. Sendo assim, o cartão de memória deverá ser colocado de acordo com o desenho localizado na parte frontal da UCP ou também como mostra a Figura 3-14.

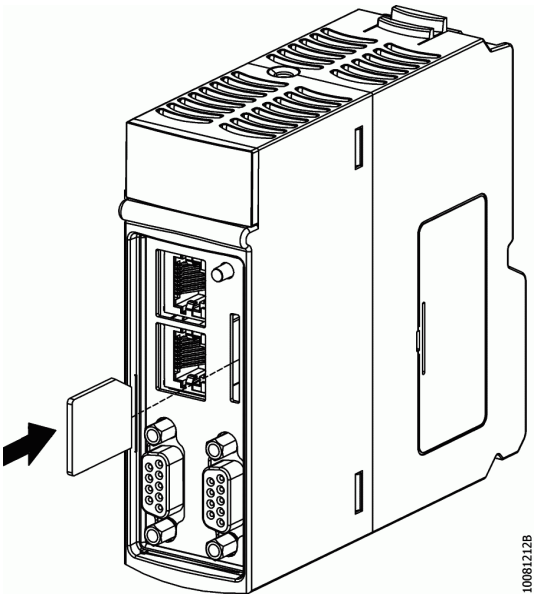

Figura 3-14. Inserção do Cartão de Memória na UCP

Quando o cartão estiver instalado corretamente aparecerá um símbolo no visor gráfico da UCP. Para remover o cartão com segurança, deve-se pressionar a tecla MS, esperar para que o símbolo do cartão desapareça do visor gráfico e então retirar o cartão. Para que isso seja possível, deve-se pressionar o cartão contra a UCP até que seja gerado um clique. Então, basta soltá-lo e retirá-lo do compartimento, conforme mostra a Figura 3-15, pois o mesmo estará solto.

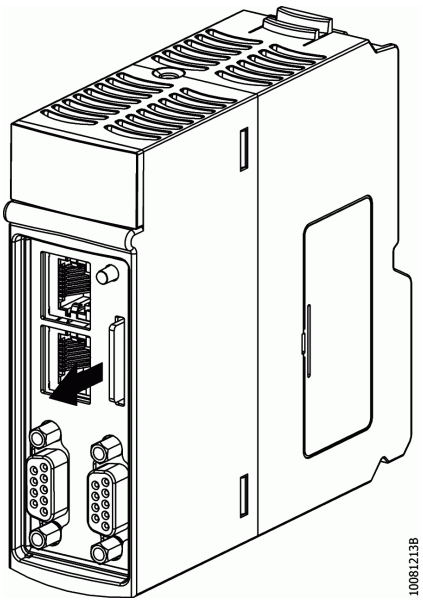

Figura 3-15. Retirada do Cartão de Memória

Instalação da Arquitetura

Instalação de Módulos no Bastidor Principal

A Série Nexto possui um exclusivo método para conectar e desconectar módulos do barramento, o qual não exige muito esforço do operador e garante a integridade da conexão. Para maiores informações sobre fixação dos produtos da Série Nexto, favor consultar o Manual de Utilização Série Nexto – MU214000.

Instalação do Programador

Para realizar a instalação do software de desenvolvimento MasterTool IEC XE é necessário ter em mãos o CD-ROM de distribuição ou efetuar o download do arquivo de instalação no site http://www.altus.com.br. Para mais informações e o passo a passo a ser seguido, consulte o Manual de Utilização MasterTool IEC XE – MU299048, no capítulo Instalação.

4. Configuração

As UCPs da Série Nexto são configuradas e programadas através do software MasterTool IEC XE. A configuração realizada define o comportamento e modos de utilização dos periféricos e características especiais das UCPs. A programação representa a aplicação desenvolvida pelo usuário, também chamada de aplicativo.

Configuração da UCP

Parâmetros Gerais

Os parâmetros relacionados abaixo fazem parte da configuração da UCP inserida na aplicação. Cada item deve ser devidamente revisado para o correto funcionamento do projeto.

Além destes parâmetros, é possível alterar o nome de cada módulo inserido na aplicação, para isto, clique com o botão direito sobre o módulo, no item "Propriedades", na guia "Comum", altere o nome, sendo limitado a 24 caracteres.

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Endereço Inicial em %Q	**Área de Diagnósticos (%Q)**		
	Endereço inicial dos diagnósticos da UCP (%Q)	Alocado automaticamente na criação do projeto.	NX3004: 0 a 32210 NX3010: 0 a 32210 NX3020: 0 a 64843 NX3030: 0 a 97611
Tamanho	Tamanho da área de diagnósticos em bytes	NX3004: 558 NX3010: 558 NX3020: 693 NX3030: 693	Não é possível alterar o tamanho da área de diagnósticos da UCP
Endereço Inicial em %Q	**Área Retentiva (%Q)**		
	Endereço inicial da memória de dados retentivos (%Q)	NX3004: 4096 NX3010: 4096 NX3020: 4096 NX3030: 4096	NX3004: 0 a 32767 menos o Tamanho da memória de dados retentivos NX3010: 0 a 32767 menos o Tamanho da memória de dados retentivos NX3020: 0 a 65535 menos o Tamanho da memória de dados retentivos NX3030: 0 a 98303 menos o Tamanho da memória de dados retentivos
Tamanho	Tamanho da memória de dados retentivos em bytes	NX3004: 2048 NX3010: 8192 NX3020: 16384 NX3030: 16384	NX3004: 0 a 2048 NX3010: 0 a 8192 NX3020: 0 a 16384 NX3030: 0 a 16384
Endereço Inicial em %Q	**Área Persistente (%Q)**		
	Endereço inicial da memória de dados persistentes (%Q)	NX3004: 12288 NX3010: 12288 NX3020: 20480 NX3030: 20480	NX3004: 0 a 32767 menos o tamanho da memória de dados persistentes NX3010: 0 a 32767 menos o tamanho da memória de dados persistentes NX3020: 0 a 65534 menos o tamanho da memória de dados persistentes NX3030: 0 a 98302 menos o tamanho da memória de dados persistentes
Tamanho	Tamanho da memória de dados persistentes em	NX3004: 2048 NX3010: 8192 NX3020: 16384	NX3004: 0 a 2048 NX3010: 0 a 8192 NX3020: 0 a 16384

	bytes.	NX3030: 16384	NX3030: 0 a 49152
Iniciar Aplicação de Usuário após Reset por Cão-de-Guarda	**Parâmetros da UCP**		
	Quando habilitado, inicia a aplicação do usuário após o reset do cão-de-guarda de hardware ou pela reinicialização do Runtime, porém mantêm a indicação do diagnóstico via LED WD e via variáveis.	Desabilitado	Habilitado Desabilitado
Troca a Quente	Modo troca a quente dos módulos.	Habilitada, sem consistência na partida.	- Desabilitada, apenas para módulos declarados - Desabilitada - Habilitada, com consistência na partida apenas para módulos declarados - Habilitada, com consistência na partida - Habilitada, sem consistência na partida
Time-out Inicial (x100 ms)	**Parâmetros TCP/IP**		
	Indica quanto tempo, depois da primeira transmissão de uma mensagem, a mesma deve ser retransmitida, assumindo que ela não foi recebida pelo destinatário. A cada retransmissão o time-out é dobrado.	2	1 a 75
Atraso no Envio do ACK (x10 ms)	Tempo de espera para envio de um comando de confirmação.	10	0 a 100
Gerar erro na consistência do cão-de-guarda das tarefas	**Parâmetros do Projeto**		
	Configuração para gerar erro na consistência do valor do cão-de-guarda, caso este seja maior que 80% do intervalo da tarefa.	Desmarcado	- Marcado: Gera erro em caso do valor do cão-de-guarda passar de 80% do valor do intervalo da tarefa - Desmarcado: Não gera erro em caso do valor do cão-de-guarda passar de 80% do valor do intervalo da tarefa.
Consiste memória endereçável retentiva e persistente	Configuração para consistir as memórias persistentes e retentivas endereçáveis.	Desmarcado	- Marcado: Consiste as memórias persistentes e retentivas endereçáveis - Desmarcado: Não consiste as memórias persistentes e retentivas endereçáveis.
Habilita atualização de E/S por tarefa	Configuração para atualizar as entradas e saídas nas tarefas em que estas são utilizadas.	Desmarcado	- Marcado: As entradas e saídas são atualizadas pelas tarefas em que forem utilizadas. - Desmarcado: As entradas e saídas são atualizadas apenas pela MainTask.

Tabela 4-1. Configurações da UCP

Notas:

Gerar erro na consistência do cão-de-guarda das tarefas: Este parâmetro foi descontinuado a partir da versão 1.32 do software MasterTool IEC XE.

Habilita atualização de E/S por tarefa: Este parâmetro foi adicionado a partir da versão 2.01 do software MasterTool IEC XE.

ATENÇÃO:
Quando o endereço inicial ou o tamanho da memória de dados retentivos ou persistentes são alterados na aplicação do usuário, a memória é totalmente realocada, fazendo com que a área de variáveis retentivas e persistentes seja limpa. Então, o usuário deverá ter precaução para não perder os dados salvos na memória.

ATENÇÃO:
Em situações em que a área de memória simbólica persistente é modificada, é apresentada uma mensagem pelo programador MasterTool IEC XE para que seja escolhido um comportamento para esta área após a carga do programa modificado. A escolha deste comportamento não afeta a área persistente de representação direta, que é sempre limpa.

ATENÇÃO:
Mesmo que um ponto de E/S seja utilizado e atualizado em outras tarefas, com a opção *Habilita atualização de E/S por tarefa* marcada, ele continuará sendo atualizado na MainTask.

Troca a Quente

As UCPs da série Nexto apresentam a possibilidade de troca dos módulos de E/S do barramento sem a necessidade de desligamento do sistema e sem perda de informações. Esta característica é conhecida como *troca a quente*.

CUIDADO:
As UCPs da Série Nexto não garantem a retentividade das variáveis persistentes e retentivas caso a fonte de alimentação, ou a própria UCP, seja removida do bastidor energizado.

Na troca a quente, o comportamento do sistema relacionado se modifica conforme a configuração definida pelo usuário, que apresenta as seguintes opções, conforme descrito abaixo:

- Desabilitada, apenas para módulos declarados
- Desabilitada
- Habilitada, com consistência na partida, apenas para módulos declarados
- Habilitada, com consistência na partida
- Habilitada, sem consistência na partida

Assim, o usuário pode escolher o comportamento que o sistema deverá apresentar em situações anormais de barramento e quando a UCP estiver em Modo Run. A Tabela 4-2 apresenta as possíveis situações anormais de barramento.

Situação	Possíveis causas
Configuração incompatível	- Algum módulo presente no barramento é diferente do modelo que está declarado na configuração
Módulo ausente	- O módulo foi retirado do barramento - Algum módulo não está respondendo à UCP por estar com defeito - Alguma posição do bastidor está com defeito

Tabela 4-2. Situações Anormais de Barramento

Para maiores informações sobre os diagnósticos correspondentes às situações descritas acima, consultar capítulo Diagnósticos via Variáveis.

Se um módulo está presente em uma posição do bastidor na qual não deveria existir módulo de acordo com a configuração, este módulo é considerado como não declarado. As opções de troca a quente *Desabilitada, apenas para módulos declarados* e *Habilitada, com consistência na partida, apenas para módulos declarados* não levam em consideração os módulos que se encontram nesta condição.

Troca a Quente Desabilitada, Apenas para Módulos Declarados

Nesta configuração, quando ocorre uma situação anormal de barramento (conforme a Tabela 4-2) a UCP entra em Modo Stop em um tempo de até 2 segundos, quando o LED DG começa a piscar 4x (conforme a Tabela 4-3). Para que a UCP volte ao estado normal Run, além de desfazer o que causou a situação anormal é necessário executar um *Reset a Quente* ou um *Reset a Frio*. Caso seja realizado um *Reset Origem*, será necessário realizar o download para que a UCP possa voltar ao estado normal Run. Os comandos de *Reset a Quente, Reset a Frio* e *Reset Origem* podem ser feito pelo MasterTool IEC XE no menu *Comunicação*.

A UCP irá permanecer no estado normal Run mesmo que encontre um módulo não declarado no barramento.

Troca a Quente Desabilitada

Esta configuração não permite qualquer situação anormal de barramento (conforme a Tabela 4-2) inclusive módulos não declarados e presentes no barramento. A UCP entra em Modo Stop , sendo que o LED DG começa a piscar 4x (conforme a Tabela 4-3). Para que, nesses casos, a UCP volte ao estado normal Run, além de desfazer o que causou a situação anormal é necessário executar um *Reset a Quente* ou um *Reset a Frio*. Caso seja realizado um *Reset Origem*, será necessário realizar o download para que a UCP possa voltar ao estado normal Run. Os comandos de *Reset a Quente, Reset a Frio e Reset Origem* podem ser feito pelo MasterTool IEC XE no menu *Comunicação*.

Troca a Quente Habilitada, com Consistência na Partida, Apenas para Módulos Declarados

É considerada "partida" o intervalo entre a energização da UCP (ou comando de reset ou download de aplicação) até a primeira vez em que a mesma entra em modo Run. Esta configuração verifica se ocorreu alguma situação anormal de barramento (conforme a Tabela 4-2) durante a partida; em caso positivo, a UCP entra em Modo Stop e o LED DG começa a piscar 4x (conforme a Tabela 4-3). Posteriormente, para que a UCP possa ser colocada em modo Run, além de corrigir o que ocasionou a situação anormal, é necessário executar um comando de *Reset a Quente* ou um *Reset a Frio*. Caso seja realizado um *Reset Origem*, será necessário realizar o download para que a UCP possa voltar ao estado normal Run. Os comandos de *Reset a Quente*, *Reset a Frio* e *Reset Origem* podem ser feito pelo MasterTool IEC XE no menu *Comunicação*.

Após a partida, se algum módulo apresentar alguma das situações citadas na tabela anterior, o sistema continuará trabalhando normalmente e sinalizará o problema via diagnóstico.

Se não existir outra situação anormal para os módulos declarados, a UCP irá para o estado normal Run mesmo que encontre um módulo não declarado no barramento.

ATENÇÃO: Nesta configuração, quando ocorrer falta de alimentação (mesmo que temporária), comando Reset a Quente, comando Reset a Frio ou ter sido realizado o Download de uma nova aplicação, e algum módulo estiver em uma situação anormal de barramento; a UCP entrará em Modo Stop e o LED DG começa a piscar 4x (conforme a Tabela 4-3), pois estas são consideradas situações de partida. Esta é a opção mais recomendada, pois garante a integridade do sistema na sua inicialização e permite a troca de módulos com o sistema funcionando.

Troca a Quente Habilitada, com Consistência na Partida

Esta configuração verifica se ocorreu alguma situação anormal de barramento (conforme a Tabela 4-2) durante a partida, inclusive se há módulos não declarados e presentes no barramento; em caso positivo, a UCP entra em Modo Stop e o LED DG começa a piscar 4x (conforme a Tabela 4-3). Posteriormente, para que a UCP possa ser colocada em modo Run, além de corrigir o que ocasionou a situação anormal, é necessário executar um comando de *Reset a Quente* ou um *Reset a Frio*. Caso seja realizado um *Reset Origem*, será necessário realizar o download para que a UCP possa voltar ao estado normal Run. Os comandos de *Reset a Quente*, *Reset a Frio* e *Reset Origem* podem ser feito pelo MasterTool IEC XE no menu *Comunicação*.

Troca a Quente Habilitada, sem Consistência na Partida

Permite que o sistema entre em operação mesmo quando algum módulo estiver em uma situação anormal de barramento (conforme Tabela 4-2). As situações anormais são relatadas via diagnóstico, tanto durante, como após a partida.

ATENÇÃO:
Esta opção é recomendada para a fase de implantação do sistema, pois permite que cargas de novas aplicações e o desligamento da alimentação sejam feitos sem a presença de todos os módulos configurados.

Como realizar a Troca a Quente

CUIDADO:
Antes de proceder à troca a quente, é importante descarregar eventuais potenciais estáticos acumulados no corpo. Para isso, toque (com as mãos nuas) em uma superfície metálica aterrada antes de manipular os módulos. Tal procedimento garante que os níveis de eletricidade estática suportados pelo módulo não serão ultrapassados.

ATENÇÃO:
Recomenda-se o monitoramento dos diagnósticos de troca a quente na aplicação de controle desenvolvida pelo usuário, a fim de garantir que o valor retornado pelo módulo seja validado antes de ser utilizado.

O procedimento para a troca de módulos a quente é descrito a seguir:

- Destrave o módulo junto ao bastidor, através da trava de segurança.
- Retire o módulo, puxando-o firmemente.
- Insira o novo módulo no bastidor.
- Certifique-se de que a trava que prende o módulo ao bastidor está totalmente conectada; caso necessário, empurre o módulo em direção ao bastidor com mais força.

No caso de módulos de saída, é conveniente que os pontos estejam desligados por ocasião da troca, a fim de reduzir a geração de arcos no conector do módulo. Isso pode ser feito pelo desligamento da fonte de campo ou pelo forçamento dos pontos via ferramentas de software. Se a carga for pequena, não há a necessidade de desligar os pontos.

É importante salientar que, nos casos em que a UCP entra em Modo Stop e o LED DG começa a piscar 4x, conforme a Tabela 4-3, devido a alguma situação anormal de barramento, conforme a Tabela 4-2; os módulos de saída tem o comportamento dos seus pontos de acordo com o que foi configurado nos Parâmetros do Módulo quando a UCP sai do Modo Run e entra em Modo Stop. Em caso de inicialização da aplicação, quando a UCP entra em Modo Stop sem ter passado para o Modo Run, os módulos de saída tem o comportamento de seus pontos em modo seguro de falha, ou seja, o ponto permanece desligado (0 Vdc).

No caso dos módulos de entrada, caso o mesmo seja removido do barramento energizado, o estado lógico dos pontos permanecerão no último valor. Caso o(s) conector(es) sejam removidos, o estado lógico dos pontos serão colocados em estado seguro, ou seja, zero ou alta impedância.

ATENÇÃO:
Proceda sempre à substituição de um módulo por vez, para que a UCP atualize os estados dos módulos.

Abaixo, a Tabela 4-3 relaciona as condições de barramento e o estado de operação e do LED DG da UCP Nexto. Para maiores informações sobre os estados dos LEDs de diagnóstico, consultar o capítulo Diagnósticos via LED.

Condição	Habilitada, com consistência na partida	Habilitada, com consistência na partida apenas para módulos declarados	Habilitada, sem consistência na partida	Desabilitada	Desabilitada, apenas para módulos declarados
Módulo não declarado	LED DG: Pisca 2x Aplicação: Run	LED DG: Pisca 2x Aplicação: Run	LED DG: Pisca 2x Aplicação: Run	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 2x Aplicação: Run
Módulo não declarado (condição partida)	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 2x Aplicação: Run	LED DG: Pisca 2x Aplicação: Run	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 2x Aplicação: Run
Módulo ausente	LED DG: Pisca 2x Aplicação: Run	LED DG: Pisca 2x Aplicação: Run	LED DG: Pisca 2x Aplicação: Run	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop
Módulo ausente (condição partida)	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 2x Aplicação: Run	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop
Configuração incompatível	LED DG: Pisca 2x Aplicação: Run	LED DG: Pisca 2x Aplicação: Run	LED DG: Pisca 2x Aplicação: Run	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop
Configuração incompatível (condição partida)	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 2x Aplicação: Run ou LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop
Endereço de slot duplicado	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop
Módulo não operacional	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop	LED DG: Pisca 4x Aplicação: Stop

Tabela 4-3. Relação entre Condições e Troca a Quente

Nota:

Habilitada, sem consistência na partida: Quando este modo de troca a quente está configurado, em situações normais quando existir um módulo incompatível na partida do sistema a aplicação passará de Stop para Run. Contudo, se o módulo configurado for um NX5000 ou NX5001e existir outro módulo na posição, a aplicação permanecerá em Stop.

Áreas de Memória Retentiva e Persistente

A UCP Nexto permite a utilização de variáveis simbólicas e de variáveis de saída de representação direta como variáveis retentivas ou persistentes.

As variáveis de saída de representação direta que serão retentivas ou persistentes devem ser declaradas nos Parâmetros Gerais da UCP, conforme descrito em Configuração - Configuração da UCP - Parâmetros Gerais. Também podem ser atribuídos nomes simbólicos a estas variáveis de saída de representação direta através da diretiva AT, Por exemplo, estando %QB4096 e %QB20480, dentro das áreas de memória retentiva e persistente, respectivamente:

```
PROGRAM MainPrg
VAR
   byVariavelRetentiva_01 AT %QB4096 : BYTE;
   byVariavelPersistente_01 AT %QB20480 : BYTE;
END_VAR
```

Caso as variáveis simbólicas que utilizam a diretiva AT não sejam declaradas dentro das áreas respectivas da memória retentiva e/ou persistente, erros durante a geração de código no MasterTool

são apresentados, informando que existem variáveis não retentivas ou não persistentes definidas em espaços de memória retentiva ou persistente.

Quanto às variáveis simbólicas que serão retentivas ou persistentes, apenas as variáveis retentivas poderão ser locais ou globais, sendo que variáveis simbólicas persistentes deverão sempre ser globais. Para a declaração de variáveis simbólicas retentivas, deve ser utilizada a palavra-chave RETAIN. Por exemplo, para variáveis locais:

```
PROGRAM MainPrg
VAR RETAIN
   wVariavelSimbolicaRetentivaLocal_01 : WORD;
END_VAR
```

Ou, para variáveis globais, declaradas dentro de uma lista de variáveis globais:

```
VAR_GLOBAL RETAIN
   wVariavelSimbolicaRetentivaGlobal_01 : WORD;
END_VAR
```

Já as variáveis simbólicas persistentes deverão ser declaradas dentro de um objeto Variáveis Persistentes, adicionado à aplicação. Estas variáveis serão globais e estarão declaradas da seguinte forma dentro do objeto:

```
VAR_GLOBAL PERSISTENT RETAIN
   wVariavelSimbolicaPersistenteGlobal_01 : WORD;
END_VAR
```

Apartir das versões 1.5.0.22 para NX3004 e 1.5.1.1 para NX3010, NX3020 e NX3030, as UCPs da série Nexto permitem flexibilidade no uso das memórias retentivas e persistentes. Isto significa que o usuário poderá optar o tamanho que irá utilizar em cada tipo de memória, desde que o somatório do tamanho das memórias retentivas e persistentes não ultrapasse o limite total disponível para cada modelo de UCP. O total de memória retentiva e persistente disponível está descrito na Tabela 2-5 em Características Específicas.

Caso o somatório de memória simbólica retentiva, simbólica persistente, retentiva em %Q e persistente em %Q ultrapassar o total disponível, o MasterTool apresentará erro durante a geração de código.

Se por exemplo, for utilizada uma UCP NX3004 que possui 7680 Bytes de memória retentiva e persistente e configurado 1000 Bytes retentivos e 1000 Bytes persistentes de saída de representação direta (%Q), e ainda, declarado as variáveis do código abaixo, o total de memória retentiva e persistente utilizada será de 2004 Bytes, restando 5676 Bytes para utilizar da forma que desejar.

```
VAR_GLOBAL PERSISTENT RETAIN
   wVariavelSimbolicaPersistenteGlobal_01 : WORD;
END_VAR

VAR_GLOBAL RETAIN
   wVariavelSimbolicaRetentivaGlobal_01 : WORD;
END_VAR
```

ATENÇÃO: Para utilizar as memórias retentivas e persistentes de forma flexível é necessário utilizar o MasterTool IEC XE 2.03 ou superior.

Configurações TCP

Algumas das configurações avançadas afetam os protocolos suportados pelas UCPs da Série Nexto, pois dizem respeito à camada de rede TCP. São elas:

- Time-out Inicial

- Atraso no envio do ACK

A UCP Nexto, antes de tratar e responder requisições e assim como qualquer outro equipamento Ethernet que utilize a camada de transporte TCP, exige a abertura de uma porta de comunicação, ou seja, do estabelecimento de uma conexão.

A quantidade de conexões da interface é limitada e, quando atingido o seu limite, a interface simplesmente não estabelece mais nenhuma conexão. Isto pode ser um problema para as conexões estabelecidas no modo servidor, pois o fechamento das conexões depende do outro equipamento, o cliente.

A camada de transporte TCP, responsável pela entrega das mensagens da origem para o destino, possui certos parâmetros envolvendo time-outs, muito comuns nos protocolos de comunicação em geral. Tais parâmetros visam à recuperação da comunicação após determinadas falhas. O usuário deve estar atento com a configuração dos time-outs, pois pode haver conflito com os valores configurados dentro da camada de aplicação, ou seja, dos protocolos. Como a configuração TCP é referência para todas as instâncias configuradas, será o tempo válido caso seja menor do que o parametrizado dentro de um protocolo:

- Time-out Inicial: indica quanto tempo, depois da primeira transmissão de uma mensagem, a mesma deve ser retransmitida, assumindo que a mesma não foi recebida pelo destinatário. A cada retransmissão o time-out é dobrado. O número de retentativas de transmissão está atrelado ao time-out da comunicação configurado dentro do protocolo, ou seja, será o tempo máximo antes de desistir da entrega da mensagem, concretizando-se então a falha de transmissão. Além disso é importante salientar que existe um limite máximo de retentativas fixo para as UCPs da Série Nexto. Este número é fixo em 1 retentativas máxima antes da conexão estabelecida e 6 retentativas após a conexão estabelecida. Toda a vez que é alcançado o número máximo de retentativas, o processo de tentativa de comunicação é reiniciado. Ver seção Configuração de Protocolos para maiores detalhes sobre a utilização desses parâmetros de time-out, pois, como descrito anteriormente, pode ser diferente dentro de cada instância. É importante ressaltar que esse parâmetro é utilizado somente no estabelecimento da conexão, após são utilizadas estatísticas das últimas comunicações para calcular o novo time-out.

Exemplo de funcionamento dos parâmetros de time-out inicial e time-out da comunicação dentro da instância MODBUS TCP Server, considerando o não recebimento de resposta de confirmação, para os seguintes valores: 300 para time-out inicial (= 300 ms) e 3000 para time-out da comunicação (= 3000 ms):

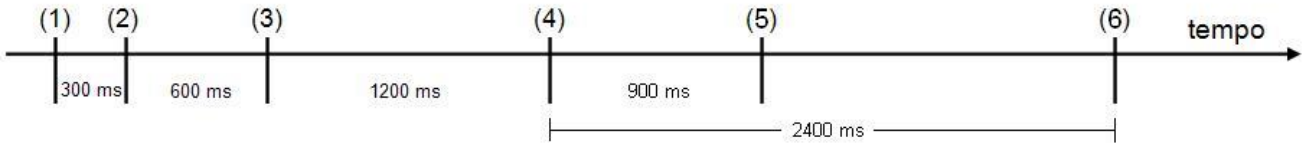

Figura 4-1. Time-out Inicial e Time-out da Comunicação

Legenda:

1. instante da transmissão da mensagem.
2. primeira retentativa de transmissão da mensagem, após o time-out inicial.
3. segunda retentativa de transmissão da mensagem, após duas vezes o time-out inicial.
4. terceira retentativa de transmissão da mensagem, após duas vezes o time-out anterior.
5. desistência de transmissão da mensagem e indicação de falha, após exceder o time-out da comunicação (tempo total até a desistência: 300 + 600 + 1200 + 900 = 3000 ms).
6. seria a quarta retentativa de transmissão da mensagem, após duas vezes o time-out anterior, porém foi excedido o time-out da comunicação configurado dentro do protocolo e a falha foi indicada.

- Atraso no Envio do ACK : define o tempo máximo aguardado pela interface para a transmissão de ACK do TCP. Este ACK é responsável pela confirmação do recebimento de uma mensagem pelo destinatário. O ajuste deste campo visa diminuir a quantidade de mensagens que circulam pela rede. A seguir, explica-se como este mecanismo funciona:
  - Todas as mensagens do tipo requisição, enviadas de um cliente para um servidor precisam ser confirmadas pelo servidor através da transmissão de uma mensagem de ACK para o cliente.
  - Todas as mensagens do tipo resposta, enviadas de um servidor para um cliente precisam ser confirmadas pelo cliente através da transmissão de uma mensagem de ACK para o servidor.
  - O não recebimento da mensagem de ACK por uma das partes, dentro do tempo hábil definido pelo parâmetro de time-out do TCP, irá ocasionar a retentativa de transmissão da mensagem por parte do endereço origem (ver parâmetro de número de retentativas do TCP).
  - A mensagem de ACK não precisa ser exclusiva, ou seja, o ACK que o servidor deve enviar para o cliente, no momento em que receber uma requisição, pode estar incluído na mesma mensagem que contém a resposta, e o ACK que o cliente deve enviar para o servidor, no momento em que receber uma resposta, pode estar incluído na mesma mensagem que contém a próxima requisição. Este conceito é conhecido como ACK calibrado.

As figuras a seguir ilustram a diferença entre o envio de um ACK imediato e um ACK calibrado:

ATENÇÃO: A UCP NX3004 possui um comportamento ligeiramente diferente, ela não considera o valor configurado no parâmetro Time out inicial, considera apenas o Time out da comunicação. Mesmo assim o valor configurado para Time out da comunicação é utilizando quando for inferior a 3 segundos e nunca será dobrado nas retentativas. Já quando o parâmetro Time out da comunicação for superior a 3 segundos, o valor é desconsiderado e passa a ser consistido o valor inicial de 3 segundos e dobrando a cada retentativa.

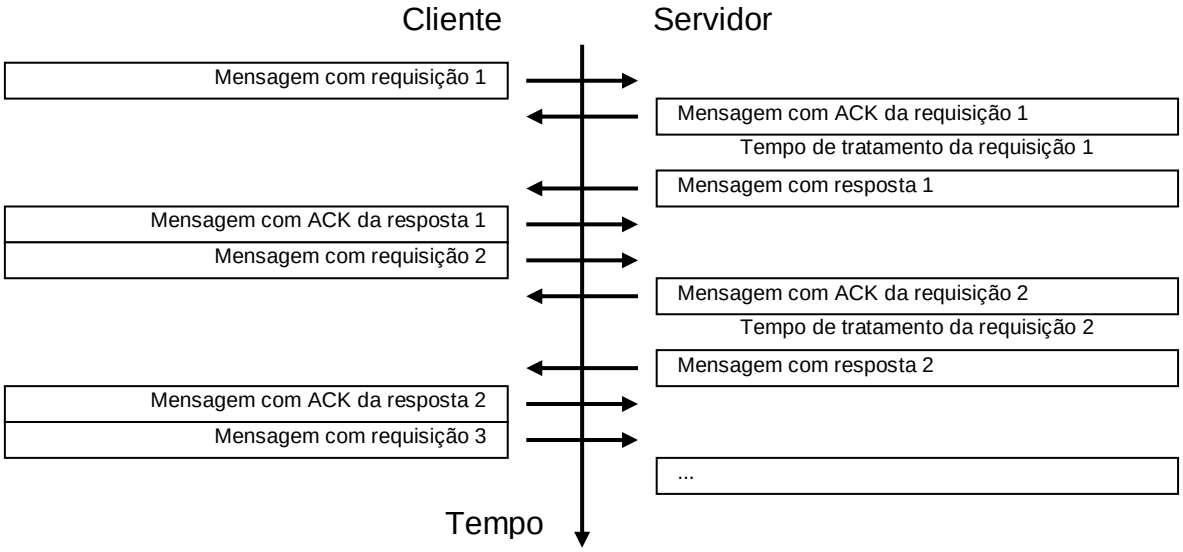
Cliente
Servidor
Mensagem com requisição 1
Mensagem com ACK da requisição 1
Tempo de tratamento da requisição 1
Mensagem com resposta 1
Mensagem com ACK da resposta 1
Mensagem com requisição 2
Mensagem com ACK da requisição 2
Tempo de tratamento da requisição 2
Mensagem com resposta 2
Mensagem com ACK da resposta 2
Mensagem com requisição 3
...
Tempo

Figura 4-2. Exemplo de Funcionamento para Envio de ACK Imediato (= 0)

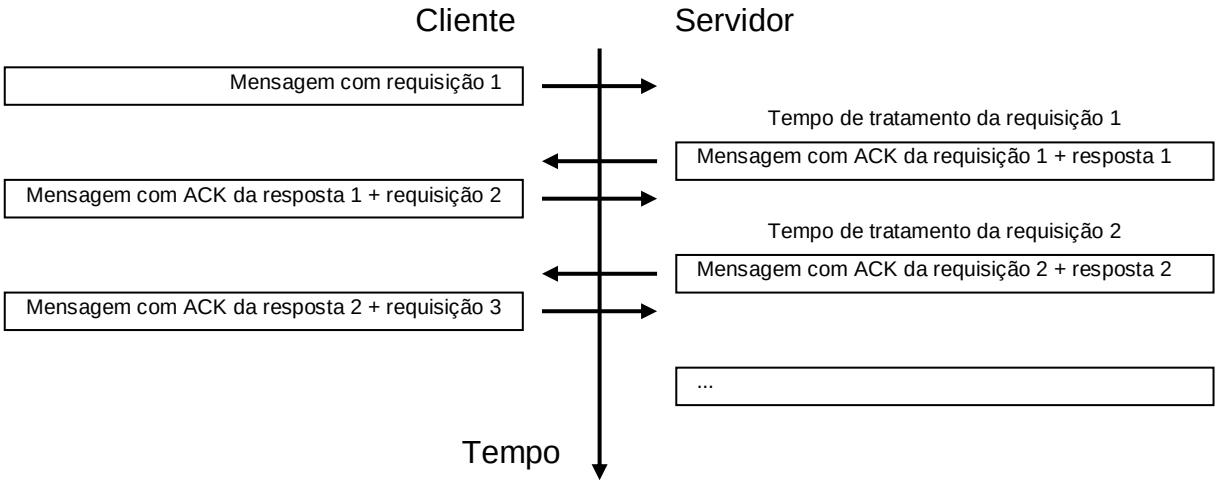
Cliente
Servidor
Mensagem com requisição 1
Tempo de tratamento da requisição 1
Mensagem com ACK da requisição 1 + resposta 1
Mensagem com ACK da resposta 1 + requisição 2
Tempo de tratamento da requisição 2
Mensagem com ACK da requisição 2 + resposta 2
Mensagem com ACK da resposta 2 + requisição 3
...
Tempo

Figura 4-3. Exemplo de Funcionamento para Envio de ACK Calibrado

ATENÇÃO:
Todos os sistemas operacionais com suporte a interfaces de rede com protocolo TCP/IP, possuem parâmetros equivalentes aos discutidos neste capítulo para a interface Ethernet das UCPs da Série Nexto. No sistema operacional Windows ®, estes parâmetros estão definidos nos registros do sistema, sob as mais distintas identificações, e devem ser modificadas apenas por administradores de rede, pois afetam todos os programas e aplicativos instalados sob o sistema operacional.

ATENÇÃO:
O parâmetro de atraso no envio do ACK se aplica somente à comunicação entre a UCP e o software MasterTool IEC XE. Para a comunicação com outros dispositivos e/ou outros protocolos (MODBUS, por exemplo) o padrão utilizado será sem atraso.

Parâmetros do Projeto

Os parâmetros do projeto da UCP são relacionadas às configurações para atualização das entradas e saídas nas tarefas em que estas são utilizadas, consistência da área retentiva e persistente em %Q e no caso das UCPs NX3010, NX3020 e NX3030, opções de gravação e leitura do cartão de memória.

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Gerar erro na consistência do cão-de-guarda das tarefas	Gera erro durante a geração de código caso alguma tarefa do projeto tenha configurado o tempo do cão-de-guarda maior que 80% do tempo configurado no Intervalo da tarefa.	Desmarcado	- Marcado - Desmarcado
Consiste área retentiva e persistente em %Q	Realiza a consistência das áreas retentivas e persistentes em %Q.	Desmarcado	- Marcado - Desmarcado
Habilita atualização de E/S por tarefa	Atualiza as entradas e saídas nas tarefas em que estas são utilizadas.	Desmarcado	- Marcado - Desmarcado
Copiar Projeto da UCP para o Cartão de Memória	**Cartão de Memória**		
	Copia o projeto da memória interna da UCP para o cartão de memória	Desabilitado	- Habilitado: Configuração habilitada - Desabilitado: Configuração desabilitada

Senha para Copiar Projeto da UCP para o Cartão de Memória	Senha para copiar o projeto da memória interna da UCP para o cartão de memória	-	Senha de 6 dígitos (0 a 999999)
Copiar Projeto do Cartão de Memória para a UCP	Copia o projeto do cartão de memória para a memória interna da UCP	Desabilitado	- Habilitado: Configuração habilitada - Desabilitado: Configuração desabilitada
Senha para Copiar Projeto do Cartão de Memória para a UCP	Senha para copiar o projeto do cartão de memória para a memória interna da UCP	-	Senha de 6 dígitos (0 a 999999)

Tabela 4-4. Parâmetros do Projeto da UCP

Notas:

Gerar erro na consistência do cão-de-guarda das tarefas: Este parâmetro foi descontinuado a partir da versão 1.32 do software MasterTool IEC XE.

Habilita atualização de E/S por tarefa: Este parâmetro foi adicionado a partir da versão 2.01 do software MasterTool IEC XE.

ATENÇÃO:
Após configurar as possibilidades de cópia do projeto e de ter criado a aplicação de inicialização, deve-se localizar o arquivo "Application.crc" para que as configurações referentes ao cartão de memória tenham efeito. A localização pode ser realizada em Selecionar o Application.crc através do botão "Localizar Arquivo...", como pode ser visto na Figura 4-66.

Configuração Evento Externo

O evento externo é uma característica disponível na UCP a qual permite que uma entrada digital, configurada pelo usuário, quando acionada, dispare a execução de uma tarefa especifica com código definido pelo usuário. Desta forma, é possível que através desta entrada, quando acionada, interrompa a execução da aplicação principal e execute a aplicação definida na tarefa ExternInterruptTask00, que possui maior prioridade que as demais tarefas de aplicação.

É importante também observar que, para evitar a geração de vários eventos em um espaço muito curto de tempo foi limitado o tratamento deste tipo de evento a cada 10 ms, ou seja, caso ocorra dois ou mais eventos durante 10 ms após o primeiro evento, o segundo e demais eventos serão descartados. Esta limitação foi imposta para evitar que um evento externo, que seja gerado de forma descontrolada, não bloqueie a UCP, visto que a sua tarefa possui maior prioridade frente às demais.

Para configurar um evento externo é necessário inserir um módulo de entrada digital, NX1001 ou NX1005, e realizar as configurações descritas a seguir, na UCP, através do software de programação MT8500.

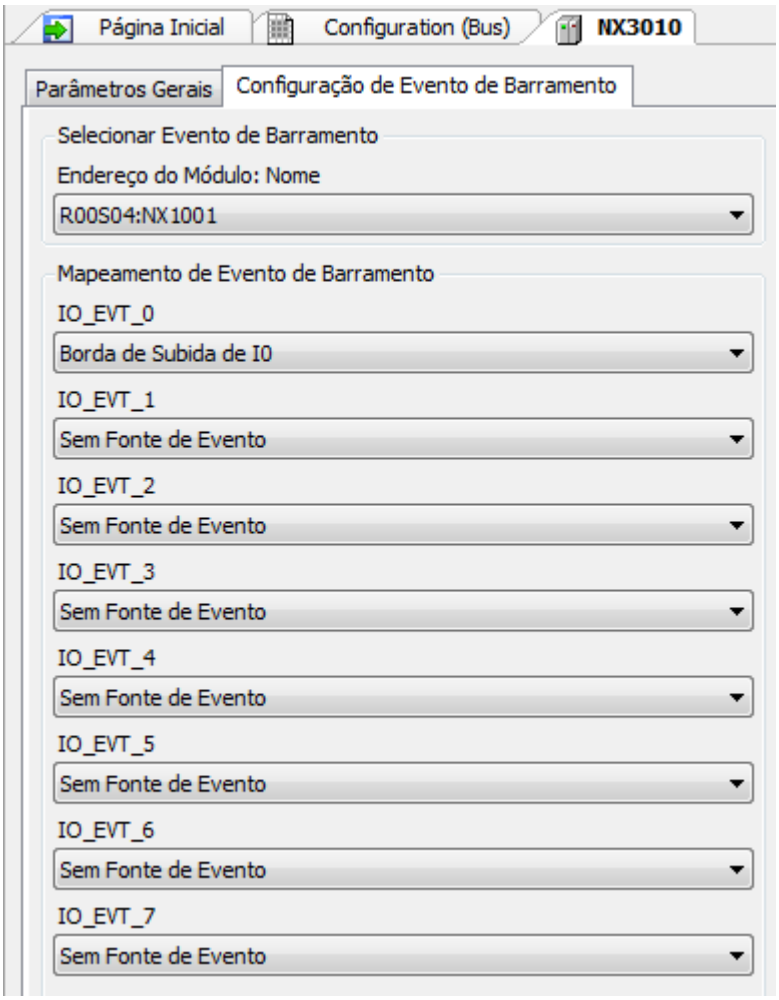

Figura 4-4. Tela de Configuração de Evento Externo na UCP

Na aba de configuração de evento externo, dentro das configurações da UCP, é preciso selecionar qual módulo será a fonte de interrupção no campo Endereço do Módulo: Nome, a seguir deve ser selecionado qual entrada deste módulo será responsável por gerar o evento (IO_EVT_0), nesta seleção é possível optar pelas opções descritas na Figura 4-5.

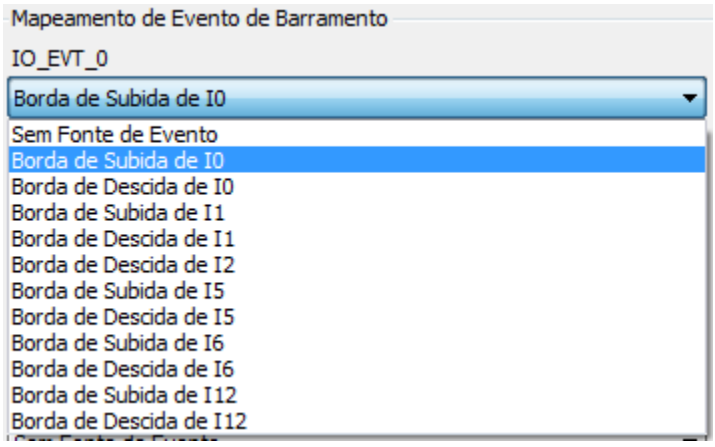
Mapeamento de Evento de Barramento
IO_EVT_0
Borda de Subida de I0
Sem Fonte de Evento
Borda de Subida de I0
Borda de Descida de I0
Borda de Subida de I1
Borda de Descida de I1
Borda de Descida de I2
Borda de Subida de I5
Borda de Descida de I5
Borda de Subida de I6
Borda de Descida de I6
Borda de Subida de I12
Borda de Descida de I12

Figura 4-5. Opções de Origem do Evento Externo no Módulo NX1001

Além da configuração da UCP é necessário configurar a tarefa responsável por executar as ações definidas pelo usuário, neste caso o usuário deve utilizar um perfil de projeto que suporte eventos externos, mais informações consulte o capítulo Perfis de Projeto. Na tela de configuração da tarefa

ExternInterruptTask00, na Figura 4-6, é necessário selecionar a origem do evento, no campo Evento, neste caso deve ser selecionado IO_EVT_0, pois as demais fontes de origem IO_EVT_1 a IO_EVT_7 não estão disponíveis. Na sequencia deve ser verificado se a POU selecionada no campo POUs está correta, pois a mesma será utilizada pelo usuário para definir as ações que serão executadas quando um evento externo ocorrer.

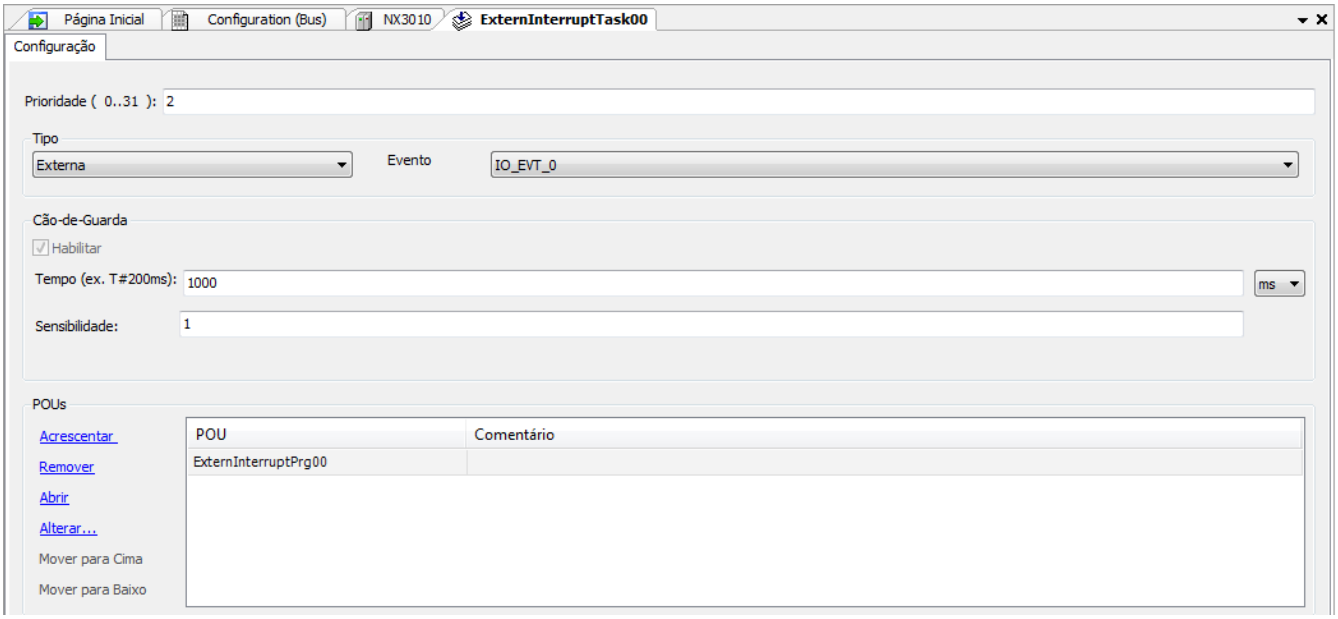

Figura 4-6. Tela de Configuração da Tarefa ExternInterruptTask00

Configuração do SOE

O SOE (Sequence of Events) é responsável pela geração de uma sequência de eventos digitais. Através do mesmo, é possível analisar o histórico comportamental das variáveis do sistema mapeadas na sua área de monitoramento. O SOE é um serviço exclusivo disponível nos modelos NX3020 e NX3030.

Uma vez habilitado o serviço SOE, a UCP passa a se comportar como um servidor DNP3, sendo assim, necessário suporte ao protocolo DNP3 por parte do cliente para utilização deste recurso. Os tipos de objetos suportados bem como os códigos de função e os qualificadores podem ser encontrados no *Anexo A*.

O serviço SOE faz a utilização de endereços %Q para constituir sua base de dados estáticos. Para isso, deve ser configurada uma área contínua de memória %Q onde o usuário irá informar seu início e tamanho dividido por dois. Para projetos redundantes a área de %Q também deve ser redundante, para que, no momento do switchover, a base de dados do servidor DNP3 seja mantida.

O endereço inicial do DNP3 sempre será 0, correspondendo ao %QBxxxx.0 e o último endereço será: ((Tamanho da área de %Q * 8) *2)- 1.

Dessa maneira, definida a base de dados estática, o usuário deve copiar cada ponto digital que deve gerar eventos para dentro da área contínua de %Q. O número máximo de pontos que podem ser copiados é 8000.

Para a configuração dos eventos, é necessário informar apenas o tamanho da fila de eventos. Essa fila é persistente e redundante, desse modo, não serão perdidos eventos no momento do switchover, ou no momento de uma falha na alimentação. Caso ocorra um overflow na fila de eventos, os eventos mais antigos serão sobrescritos. Caso em um único ciclo sejam gerados mais eventos do que o suportado pela fila, sua geração é interrompida e o diagnóstico de overflow é ligado (SOE[x].bOverflowStatus). Por exemplo, se 100+n bits variam em uma configuração de 100 eventos, ocorre um descarte de n eventos.

O SOE irá rodar no contexto da tarefa MainTask, já a partir do primeiro ciclo. O SOE irá ser executado ao final de cada ciclo da tarefa e comparará os bits mapeados, a fim de detectar transições ocorridas durante o ciclo. Sendo assim, a cada ciclo em que eventos sejam gerados, irá ocorrer um acréscimo de tempo neste ciclo da MainTask. Para o pior caso (1000 eventos, sendo que serão gerados apenas 1000 e o restante descartado), esta influência será de aproximadamente 5 ms. Portanto, para uma aplicação com o SOE habilitado, o usuário deverá levar em consideração este tempo ao configurar os parâmetros de tempo de cão-de-guarda e intervalo da tarefa MainTask.

Para a utilização do mesmo o usuário deverá configurar os seguintes parâmetros na aba Configuração do SOE:

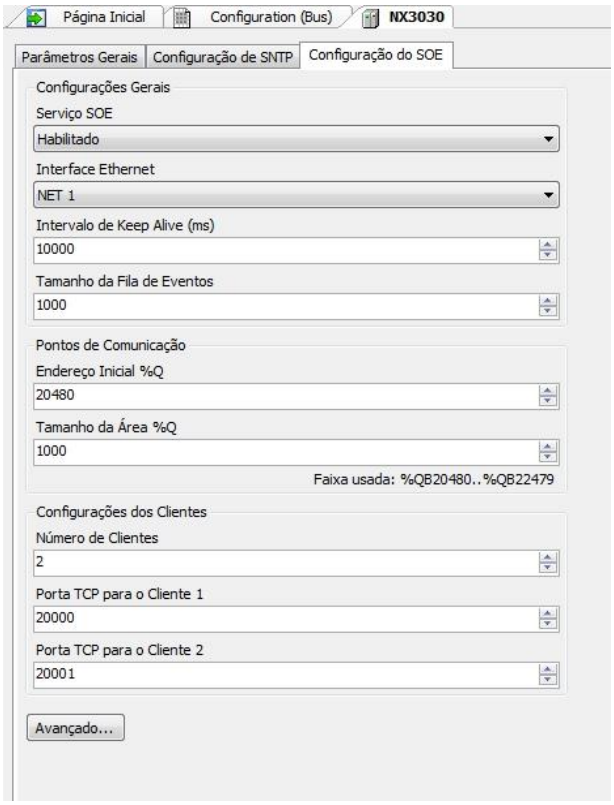

Figura 4-7. Configuração da Sequência de Eventos

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Configurações Gerais			
Serviço SOE	Ativa o SOE.	Desabilitado	Habilitado Desabilitado
Interface Ethernet	Seleciona a interface utilizada.	NET 1	NET 1 NET 2
Intervalo para Keep Alive	Intervalo das mensagens de keep alive (ms).	10000	0 a 4294967295
Tamanho da Fila de Eventos	Tamanho da fila de eventos.	1000	100 a 1000
Pontos de Comunicação			
Endereço Inicial dos Dados em %Q	Endereço inicial para dados estáticos.	20480	Pode ser utilizado qualquer endereço da área de %Q
Tamanho	Tamanho da memória a ser utilizada pelos dados estáticos (%Q).	1000	1 a 1000
Configuração dos Clientes			
Configuração	**Descrição**	**Padrão de Fábrica**	**Possibilidades**
Número de Clientes	Define o número de clientes.	2	1, 2
Porta TCP do Cliente 1	Seleciona a porta de comunicação para o primeiro cliente.	20000	1 a 65535
Porta TCP do Cliente 2	Seleciona a porta de comunicação para o segundo cliente.	20001	1 a 65535

Tabela 4-5. Configuração do SOE

Notas:

Tamanho da Memória de Dados: O tamanho da área de memória reservada para ser utilizada pelos dados estáticos será sempre o dobro do valor configurado, pois a segunda metade da área de memória é utilizada para armazenar os valores anteriores das variáveis da primeira metade.

Keep Alive: Enquanto estiver conectado com um cliente, serão enviadas mensagens de keep alive em intervalos, conforme o configurado. Caso o cliente não responda a estas mensagens, a conexão é fechada. Ou seja, uma conexão entre cliente e servidor poderá levar um tempo igual ao intervalo configurado para ser fechada em caso de erro.

Nas opções avançadas (botão Avançado...) é possível configurar os endereços de comunicação referentes ao protocolo DNP3.

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Endereço DNP3 da Origem	Endereço de Origem (CP)	4	0 a 65519
Endereço DNP3 do Destino do Cliente 1	Endereço do primeiro cliente	3	0 a 65519
Endereço DNP3 do Destino do Cliente 2 DNP3	Endereço do segundo cliente	3	0 a 65519

Tabela 4-6. Configurações Avançadas do SOE

Nota:

Endereço DNP3: Os endereços DNP3 da faixa de 65520 a 65535 não podem ser configurados na origem ou em um destino, pois os mesmos são utilizados para mensagens em broadcast.

ATENÇÃO:
As mensagens de Data Link do DNP3 não são utilizadas pelas UCPs da série Nexto, pois a norma não recomenda o seu uso ao comunicar por TCP/IP.

Sincronização de Tempo

Para a sincronização de tempo, as UCPs da série Nexto usam o protocolo SNTP (Simple Network Time Protocol). Para isso, elas se comportarão como um cliente SNTP, ou seja, enviando requisições de sincronização de tempo para um servidor SNTP/NTP, que pode estar na rede local ou na internet. O cliente SNTP trabalha com resolução de 1 ms, com 100 ms de precisão, ou seja, quando uma sincronização for efetuada, o horário atualizado no cliente pode estar até 100 ms adiantado ou atrasado em relação ao servidor.

A UCP envia as requisições de sincronização cíclicas, de acordo com o tempo configurado no campo Período de Sincronização do SNTP. Na primeira tentativa de sincronização, logo após a inicialização do serviço, a requisição é para o primeiro servidor, configurado em Endereço de IP do 1º Servidor. Caso este não responda, as requisições são direcionadas para o segundo servidor configurado em Endereço de IP do 2º Servidor, fornecendo uma redundância de servidores SNTP. Caso o segundo servidor também não responda, o mesmo processo de tentativa de sincronização é executado novamente, mas apenas após o Período de Sincronização ter passado. Ou seja, a cada período de sincronização, a UCP tenta se conectar uma vez em cada servidor, ela tenta o segundo caso o primeiro não responda. O tempo de espera por uma resposta do servidor SNTP é definido por padrão em 5 segundos e não pode ser modificado.

Caso, após uma requisição de sincronização, a diferença entre o horário atual da UCP e o recebido pelo servidor for maior que o valor configurado no parâmetro *Erro Mínimo Antes da Atualização do Relógio*, o horário da UCP é atualizado.

O SNTP usa o horário no formato UTC (Universal Time Coordinated), logo o parâmetro de Fuso Horário deve ser configurado corretamente, para que o horário lido pelo SNTP seja convertido corretamente para a hora local.

O Processo de execução do cliente SNTP pode ser exemplificado com os seguintes passos:

1. Tentativa de sincronização através do primeiro servidor. Caso a sincronização ocorra com sucesso, a UCP aguarda o tempo para a nova sincronização (Período de Sincronização) e tentará sincronizar-se novamente com o este servidor, utilizando, então, este como servidor primário. Em caso de falha (o servidor não responde em menos de 5 s) o passo 2 é executado.
2. Tentativa de sincronização através do segundo servidor. Caso a sincronização ocorra com sucesso, a UCP aguarda o tempo para a nova sincronização (Período de Sincronização) e tentará sincronizar-se novamente com este servidor, utilizando, então, este como servidor primário. Em caso de falha (o servidor não responde em menos de 5 s) é aguardado o tempo referente ao Período de Sincronização e executado novamente o passo 1.

Como o tempo de espera pela resposta do servidor SNTP é de 5 s, deve-se prestar atenção ao configurar valores menores do que 10 s para o Período de Sincronização. Caso o servidor primário não responda, o tempo para a sincronização irá ser de, no mínimo, 5 s (aguardo da resposta do servidor primário e tentativa de sincronização com o servidor secundário). Caso nem o servidor primário nem o secundário respondam, o tempo para a sincronização irá ser de, no mínimo, 10 s (aguardo da resposta dos dois servidores e nova tentativa de conexão com o 1º servidor).

Quando utilizado as UCPs NX3020 ou NX3030, dependendo da subrede do servidor SNTP, o cliente irá utilizar a interface Ethernet que esteja na subrede corresponde para fazer as requisições de sincronismo. Caso não haja uma interface configurada na mesma subrede do servidor, a requisição poderá ser feita por qualquer interface que possa achar uma rota para o servidor, sendo isso também válido para os modelos NX3004 e NX3010.

Para a utilização do cliente SNTP, o usuário deve configurar os seguintes parâmetros na aba Configuração de SNTP, acessado através da UCP, na árvore de dispositivos:

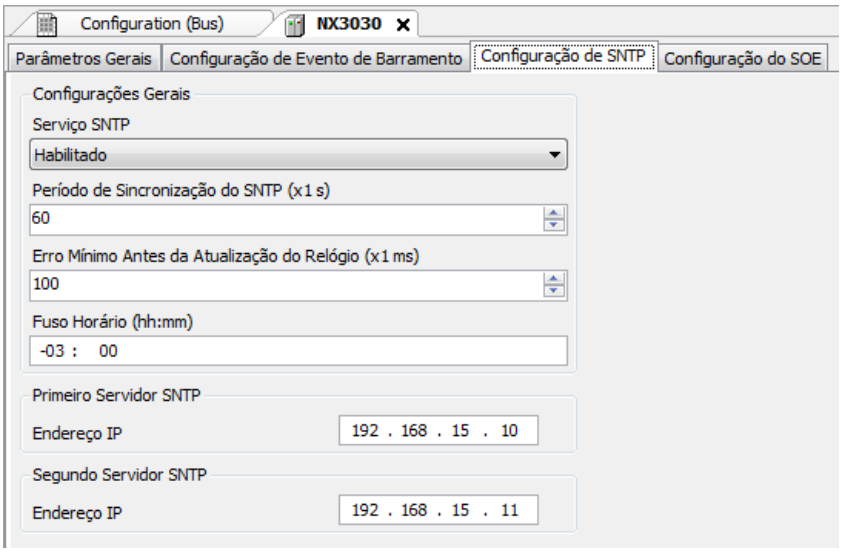

Figura 4-8. Configuração de SNTP

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Serviço SNTP	Ativa o serviço SNTP.	Desabilitado	Desabilitado Habilitado
Período de Sincronização do SNTP (x1 sec)	Intervalo de tempo das requisições de sincronização (segundos).	60	1 a 255
Erro Mínimo Antes da Atualização do Relógio (x1 ms)	Valor de offset aceitável entre a hora do servidor e do cliente (milissegundos).	100	100 a 65519
Fuso Horário (hh:mm)	Fuso horário de sua localidade. Pode ser inserida a hora e o minuto.	-3:00	-12:59 a +13:59
Endereço IP do 1º Servidor SNTP	Endereço IP do servidor SNTP primário.	192.168.15.10	1.0.0.0 a 223.255.255.255
Endereço IP do 2º Servidor SNTP	Endereço IP do servidor SNTP Secundário.	192.168.15.11	1.0.0.0 a 223.255.255.255

Tabela 4-7. Configurações SNTP

Notas:

Servidor SNTP: É possível definir um endereço preferencial e outro secundário para acessar um servidor SNTP e, assim, obter sincronismo de tempo. Se ambos os campos estiverem vazios, o serviço SNTP permanecerá desabilitado.

Padrão de Fábrica: A partir da versão de MasterTool 1.40 e superiores do MasterTool IEC XE o valor Padrão de Fábrica para os Endereços IP dos Servidores SNTP foram alterados.

ATENÇÃO:
O Serviço SNTP depende da aplicação do usuário apenas para a sua configuração. Portanto, este serviço vai ser executado mesmo quando a UCP estiver nos modos STOP ou BREAKPOINT, desde que exista uma aplicação na UCP com o cliente SNTP habilitado e corretamente configurado.

CUIDADO:
É vital a configuração de, no mínimo, um servidor SNTP. O recomendável é configurar dois servidores SNTP (primário e secundário). O sincronismo SNTP é necessário para gerar eventos com estampa de tempo coerente entre os CPA e CPB e com a hora mundial. Outra utilidade é evitar descontinuidades durante um switchover em aplicações que referenciam data e hora, tendo em vista que não existe sincronismo de data e hora entre os CPs através dos canais de sincronismo NETA e NETB.

Horário de Verão

A configuração do horário de verão deve ser feita indiretamente, através da função SetTimeZone, que altera o fuso horário aplicado ao RTC. No início do horário de verão, deve-se usar a função para aumentar em uma hora o fuso horário. Ao final do horário de verão, ela é usada novamente para diminui-lo em uma hora.

Para maiores informações, consultar a subseção Relógio RTC.

Configuração das Interfaces Seriais

COM 1 (NX3010/NX3020/NX3030)

A interface de comunicação COM 1, presente na UCPs NX3010, NX3020 e NX3030, é composta por um conector tipo DB9 fêmea para o padrão RS-232C. Permite a comunicação ponto a ponto (ou em rede utilizando conversor) nos protocolos abertos, MODBUS RTU escravo ou MODBUS RTU mestre.

Abaixo, segue os parâmetros que devem ser configurados para o bom funcionamento da aplicação.

Quando utilizado o protocolo MODBUS Mestre/Escravo, alguns destes parâmetros (como Modo Serial, Bits de Dado, Threshold RX e Eventos Seriais) são ajustados automaticamente pela ferramenta MasterTool para o correto funcionamento deste protocolo.

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Tipo Serial	Configuração do tipo do canal serial.	RS-232C	RS-232C
Taxa de Transmissão	Velocidade da porta de comunicação serial.	115200	200, 300, 600, 1200, 1800, 2400, 4800, 9600, 19200, 38400, 57600, 115200 bps
Paridade	Configura a paridade da porta serial.	Sem Paridade	Ímpar Par Paridade sempre Um Paridade Sempre Zero Sem Paridade
Bits de Dado	Configura o número de bits de dados em cada caractere da comunicação serial.	8	5, 6, 7 e 8
Bits de Parada	Configura os bits de parada da porta serial.	1	1, 1.5 e 2
Modo Serial	Configura o modo de operação da porta serial.	Modo Normal	- Modo Estendido: Modo estendido de operação da comunicação serial, no qual são fornecidas informações sobre o frame de dados recebido. - Modo Normal: Modo normal de operação da comunicação serial

Tabela 4-8. Configurações da Interface Serial Padrão RS-232C

Notas:

Modo Estendido: Neste modo de operação da comunicação serial são fornecidas informações sobre o frame de dados recebido. As informações disponibilizadas são as seguintes:

Um byte para o dado recebido (RX_CHAR : BYTE): Armazena os cinco, seis, sete ou oito bits dos dados recebidos, dependendo da configuração da serial.

Um byte para os erros no sinal (RX_ERROR : BYTE): Tem o seguinte formato:

- Bit 0: 0 - o caractere nos bits 0 a 7 é válido. 1 - o caractere nos bits 0 a 7 não é válido (ou pode não ser válido), devido aos problemas indicados nos bits 10 a 15.
- Bit 1: Não utilizado.
- Bit 2: Não utilizado.
- Bit 3: Erro de interrupção na UART. A entrada serial permaneceu na lógica 0 (paridade sempre zero) por um tempo maior do que um caractere (bit de partida + bit de dados + bit de paridade + bit de parada).
- Bit 4: Erro no frame UART. A lógica 0 (paridade sempre zero) foi lida quando o primeiro bit de parada era esperado, sendo que deveria ser lógica 1 (paridade sempre um).
- Bit 5: Erro de paridade UART. O bit de paridade lido não está de acordo com o bit de paridade calculado.
- Bit 6: Erro de overrun UART. Dados foram perdidos durante a leitura do FIFO UART, porque novos caracteres foram recebidos antes de os antigos serem removidos. Esse erro somente será indicado no primeiro caractere lido após a indicação do erro de overrun. Isso significa que alguns dados antigos foram perdidos.
- Bit 7: Erro de overrun na fila RX: Esse caractere foi escrito quando a fila RX foi completa, sobrescrevendo caracteres não lidos.

Dois bytes para o sinal de timestamping (RX_TIMESTAMP : WORD): Indica o tempo de silêncio, no intervalo de 0 a 65535, usando como base de tempo 10 us. Satura em 655,35 ms, caso o tempo de silêncio seja maior do que 65535 unidades. O RX_TIMESTAMP de um caractere mede o tempo desde uma referência que pode ser uma das três opções abaixo:

- Na maioria dos casos, o final do caractere anterior.
- Configuração da porta serial.
- O fim da transmissão serial usando o SERIAL_TX FB, ou seja, quando o último caractere foi enviado na linha.

Além de medir o tempo de silêncio antes de cada caractere, o RX_TIMESTAMP também é importante, pois mede o tempo de silêncio do último caractere da fila RX.

A medição do silêncio é importante para implementar corretamente alguns protocolos, como MODBUS RTU. Esse protocolo especifica um interframe maior do que 3,5 caracteres e um inter-byte menor do que 1,5 caracteres.

Bits de Dado: A configuração Bits de Dado das interfaces seriais limita os campos Bits de Parada e Paridade da comunicação, ou seja, o número de bits de parada e o método de paridade irão variar de acordo com o número de bits de dado. Abaixo, a Tabela 4-9 mostra as configurações permitidas para a interface COM 1das UCPs NX3010, NX3020 e NX3030.

Bits de Dados	Bits de Parada	Paridade
5	1,1.5	SEM PARIDADE, ÌMPAR, PAR, PARIDADE SEMPRE UM, PARIDADE SEMPRE ZERO
6	1, 2	SEM PARIDADE, ÌMPAR, PAR, PARIDADE SEMPRE UM, PARIDADE SEMPRE ZERO
7	1, 2	SEM PARIDADE, ÌMPAR, PAR, PARIDADE SEMPRE UM, PARIDADE SEMPRE ZERO
8	1, 2	SEM PARIDADE, ÌMPAR, PAR, PARIDADE SEMPRE UM, PARIDADE SEMPRE ZERO

Tabela 4-9. Configurações Específicas

Configurações Avançadas

As configurações avançadas são relacionadas aos sinais de controle da comunicação serial, ou seja, quando se faz necessária a utilização de um controle mais apurado da transmissão e recepção dos dados.

Configuração	Descrição	Padrão de Fábrica	Possibilidades

Parâmetros de Porta Avançados			
Sinais de modem	Realiza o controle de requisição para transmissão de um comando, através da interface RS-232C.	RTS Sempre Desligado	- RTS: Habilitado no início da transmissão e reiniciado, o mais rápido possível, após o final da transmissão. Ex: Controle de um conversor RS-232C/RS-485 externo. - RTS Sempre Desligado: Sempre desabilitado. - RTS Sempre Ligado: Sempre habilitado. - RTS/CTS: Caso o CTS esteja desabilitado, o RTS é habilitado. Então, aguarda-se o CTS ser habilitado para a transmissão começar e o RTS é reiniciado, o mais rápido possível, no final da transmissão. Ex: Controle de rádio modems com o mesmo sinal de modem. - RTS Manual: o usuário é responsável por controlar todos os sinais de controle.
Threshold de RX da UART	Quantidade de bytes que devem ser recebidos para gerar uma nova interrupção na UART. Valores baixos fazem o TIMESTAMP mais preciso quando o MODO ESTENDIDO é utilizado e minimiza os erros de overrun. No entanto, valores baixos podem causar muitas interrupções que podem retardar a UCP.	8	1, 4, 8 e 14
Eventos Seriais			
RX em TX	Quando verdadeiro, todos os bytes recebidos durante a transmissão serão descartados em vez de ir para a fila de RX. Utilizado para desabilitar a operação full-duplex na interface RS-232C.	Desabilitado	- Habilitado: Configuração habilitada - Desabilitado: Configuração desabilitada
Evento RX DCD	Quando verdadeiro, gera um evento externo devido à mudança do sinal DCD.	Habilitado	- Habilitado: Configuração habilitada - Desabilitado: Configuração desabilitada
Evento RX CTS	Quando verdadeiro, gera um evento externo devido à mudança do sinal CTS.	Habilitado	- Habilitado: Configuração habilitada - Desabilitado: Configuração desabilitada

Tabela 4-10. Configurações Avançadas da Interface Serial Padrão RS-232C

Notas:

RX em TX: Este parâmetro avançado é valido para configurações RS-232C e para RS-422.

Evento RX DCD: Os eventos externos, como o sinal de DCD da COM 1 das UCPs NX3010, NX3020 e NX3030, só podem ser associados às tarefas do perfil de projeto personalizado, descrito com detalhes no Manual de Manual de Utilização do MasterTool IEC XE – MU299048.

Evento RX CTS: Os eventos externos, como o sinal de CTS da COM 1 das UCPs NX3010, NX3020 e NX3030, só podem ser associados às tarefas do perfil de projeto personalizado, descrito com detalhes no Manual de Manual de Utilização do MasterTool IEC XE – MU299048.

COM 1 (NX3004) e COM 2 (NX3010/NX3020/NX3030)

As interfaces de comunicação COM 1 (NX3004) e COM 2 (NX3010, NX3020 e NX3030) são compostas por um conector tipo DB9 fêmea para os padrões RS-422 e RS-485. Permitindo a comunicação ponto a ponto ou em rede nos protocolos abertos, MODBUS RTU Escravo ou MODBUS RTU Mestre.

Quando utilizado o protocolo MODBUS Mestre/Escravo, alguns destes parâmetros (como Modo Serial, Bits de Dado, Threshold RX e Eventos Seriais) são ajustados automaticamente pela ferramenta MasterTool para o correto funcionamento deste protocolo.

Abaixo, segue os parâmetros que devem ser configurados para o bom funcionamento da aplicação.

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Tipo Serial	Configuração do tipo do canal serial.	RS-485	RS-422 e RS-485
Taxa de Transmissão	Velocidade da porta de comunicação serial.	115200	200, 300, 600, 1200, 1800, 2400, 4800, 9600, 19200, 38400, 57600, 115200 bps
Paridade	Configura a paridade da porta serial.	Sem Paridade	Ímpar Par Paridade Sempre Um Paridade Sempre Zero Sem Paridade
Bits de Dado	Configura o número de bits de dados em cada caractere da comunicação serial.	8	5, 6, 7 e 8
Bits de Parada	Configura os bits de parada da porta serial.	1	1, 1.5 e 2
Modo Serial	Configura o modo de operação da porta serial.	Modo Normal	- Modo Estendido: Modo estendido de operação da comunicação serial, no qual são fornecidas informações sobre o frame de dados recebido (ver nota seção COM 1). - Modo Normal: Modo normal de operação da comunicação serial

Tabela 4-11. Configuração da Interface Serial Padrões RS-422 e RS-485

A configuração Bits de Dado das interfaces seriais limita os campos Bits de Parada e Paridade da comunicação, ou seja, o número de bits de parada e o método de paridade irá variar de acordo com o número de bits de dado. Abaixo, a Tabela 4-12 mostra as configurações permitidas para as interfaces COM 1 (NX3004) e COM 2 (NX3010, NX3020 e NX3030).

Bits de Dados	Bits de Parada	Paridade
5	1, 1.5	Sem Paridade, Ímpar, Par, Paridade Sempre Um, Paridade Sempre Zero
6	1, 2	Sem Paridade, Ímpar, Par, Paridade Sempre Um, Paridade Sempre Zero
7	1, 2	Sem Paridade, Ímpar, Par, Paridade Sempre Um, Paridade Sempre Zero
8	1, 2	Sem Paridade, Ímpar, Par, Paridade Sempre Um, Paridade Sempre Zero

Tabela 4-12. Configurações Específicas

Configurações Avançadas

As configurações avançadas são relacionadas aos sinais de controle da comunicação serial, ou seja, quando se faz necessária a utilização de um controle mais apurado da transmissão e recepção dos dados.

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Threshold de RX da UART	Quantidade de bytes que devem ser recebidos para gerar uma nova interrupção na UART. Valores baixos fazem o TIMESTAMP mais preciso quando o MODO ESTENDIDO é utilizado e minimiza os erros de overrun. No entanto, valores baixos podem causar muitas interrupções que podem retardar a UCP.	8	1, 4, 8 e 14

Tabela 4-13. Configurações Avançadas da Interface Serial Padrões RS-422C e RS-485

Configuração das Interfaces Ethernet

As UCPs Nexto podem disponibilizar até duas interfaces Ethernet locais, NET 1 e NET 2. As UCPs NX3004 e NX3010 possuem apenas a interface NET 1 e as UCPs NX3020 e NX3030 possuem NET 1 e NET 2. Além das interfaces Ethernet locais, a Série Nexto disponibiliza também interfaces Ethernet remotas através da inclusão do módulo NX5000. Os módulos NX5000 possuem apenas a interface NET 1.

Interfaces Ethernet Locais

NET 1

A interface NET 1 é composta por um conector tipo RJ45 de comunicação no padrão 10/100Base-TX. Permite a comunicação ponto a ponto ou em rede nos protocolos abertos, como por exemplo, MODBUS TCP Cliente, MODBUS RTU via TCP Cliente, MODBUS TCP Servidor e MODBUS RTU via TCP Servidor.

Abaixo, segue os parâmetros que devem ser configurados para o bom funcionamento da aplicação.

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Endereço IP	Endereço IP do Controlador no barramento Ethernet.	192.168.15.1	1.0.0.1 a 223.255.255.254
Máscara de Subrede	Máscara de subrede do Controlador no barramento Ethernet.	255.255.255.0	128.0.0.0 a 255.255.255.252
Endereço do Gateway	Endereço do Gateway do Controlador no barramento Ethernet.	192.168.15.253	1.0.0.1 a 223.255.255.254

Tabela 4-14. Configurações NET 1

NET 2

A interface NET 2 é composta por um conector tipo RJ45 de comunicação no padrão 10/100Base-TX. Permite a comunicação ponto a ponto ou em rede nos protocolos abertos, como por exemplo, MODBUS TCP Cliente, MODBUS RTU via TCP Cliente, MODBUS TCP Servidor e MODBUS RTU via TCP Servidor.

Abaixo, segue os parâmetros que devem ser configurados para o bom funcionamento da aplicação.

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Endereço IP	Endereço IP do Controlador no barramento Ethernet.	192.168.16.1	1.0.0.1 a 223.255.255.254
Máscara de Subrede	Máscara de subrede do Controlador no barramento Ethernet.	255.255.255.0	128.0.0.0 a 255.255.255.252
Endereço do Gateway	Endereço do Gateway do Controlador no barramento Ethernet.	192.168.16.253	1.0.0.1 a 223.255.255.254

Tabela 4-15. Configurações NET 2

ATENÇÃO:
Não é possível configurar as duas interfaces Ethernet de uma UCP na mesma subrede, sendo este tipo de configuração bloqueada pela ferramenta MasterTool. Desta forma, cada interface Ethernet deve ser configurada em uma subrede distinta.

Interfaces Ethernet Remotas

NET 1

A interface NET 1 é composta por um conector tipo RJ45 de comunicação no padrão 10/100Base-TX. Permite a comunicação ponto a ponto ou em rede nos protocolos abertos, MODBUS TCP Cliente, MODBUS RTU via TCP Cliente, MODBUS TCP Servidor e MODBUS RTU via TCP Servidor.

Abaixo, segue os parâmetros que devem ser configurados para o bom funcionamento da aplicação.

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Endereço IP	Endereço IP do Controlador no barramento Ethernet.	192.168.xx.68	1.0.0.1 a 223.255.255.254
Máscara de Subrede	Máscara de subrede do Controlador no barramento Ethernet.	255.255.255.0	128.0.0.0 a 255.255.255.252
Endereço do Gateway	Endereço do Gateway do Controlador no barramento Ethernet.	192.168.xx.253	1.0.0.1 a 223.255.255.254

Tabela 4-16. Configurações NET 1 Remota

Portas TCP Reservadas

As seguintes portas TCP das interfaces Ethernet, tanto locais quanto remotas, são utilizadas por serviços da UCP e, portanto, são reservadas e não devem ser utilizadas pelo usuário: 80, 161, 8080, 1217, 1740, 1741, 1742,1743 e 11740.

Configuração do Módulo NX5000

Os módulos NX5000 podem ser inseridos no projeto para aumentar a quantidade de interfaces Ethernet caso as interfaces locais da UCP não sejam suficientes. Apenas as UCPs NX3020 e NX3030 tem suporte ao módulo NX5000.

Os canais Ethernet dos módulos NX5000 podem ser usados de forma individual, ou organizados em pares NIC Teaming. Pares NIC Teaming são utilizados para prover canais Ethernet redundantes.

Uma aplicação típica para o módulo NX5000 pode ser construir uma HSDN (High Speed Deterministic Network), redundante, para comunicação entre diversos CPs. Através desta rede, diversos CPs podem trocar mensagens em uma rede totalmente segregada, para garantir determinismo e uma comunicação rápida. Além disso, configurando esta rede como redundante com pares NIC Teaming, pode-se obter excelente disponibilidade. Para construir uma HSDN redundante,

deve-se inserir dois módulos NX5000. A Figura 4-9 mostra um exemplo de HSDN redundante, utilizando dois módulos NX5000.

A Figura 4-9 também mostra um exemplo com um módulo NX5000 usado de forma isolada (sem redundância NIC Teaming), inserido à direita dos demais módulos.

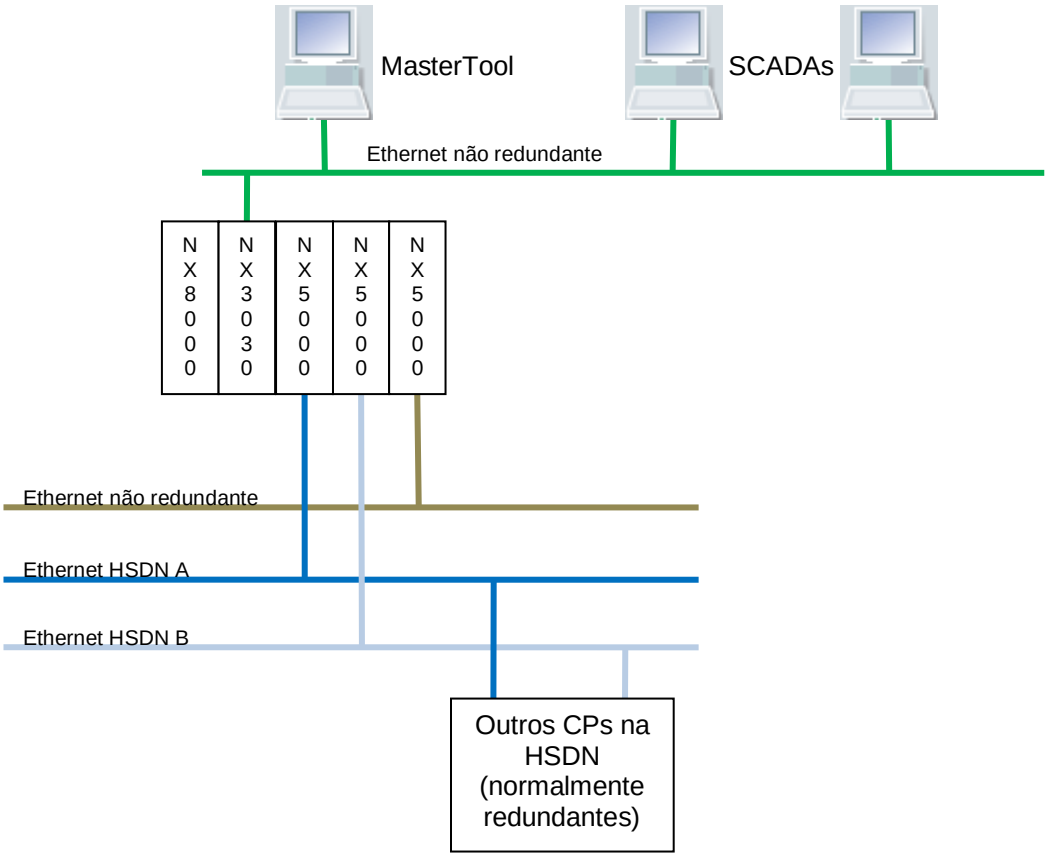

Figura 4-9. Redes Ethernet Simples e Redundante Utilizando NX5000

Os dois primeiros NX5000 do bastidor formam um par redundante NIC Teaming, interligados em dois switches diferentes (Ethernet HSDN A e Ethernet HSDN B). Em algum ponto, estes dois switches devem ser interligados, para que exista conexão entre as duas portas NIC Teaming e disponibilidade ainda maior (contra falhas duplas).

Tais arquiteturas Ethernet possibilitam excelente disponibilidade, contra falha nas portas Ethernet, em cabos e em switches.

Um conjunto de duas portas Ethernet formando um par NIC Teaming possui um único endereço de IP, vinculado ao par de portas. Desta forma, um cliente como um SCADA ou MasterTool, conectado a um servidor no CP, não precisa se preocupar em trocar o endereço IP caso haja falha em algumas das portas do par NIC Teaming.

Diagnósticos apontam falhas que eventualmente surjam em qualquer das portas de um par NIC Teaming.

ATENÇÃO: Ambas as UCPs NX3020 e NX3030 tem suporte ao módulo NX5000, podendo-se, inclusive, agrupar dois módulos NX5000 como um par NIC Teaming.

Utilizando a UCP NX3020 podem ser inseridos no projeto até dois módulos NX5000, e utilizando a UCP NX3030 podem ser inseridos até seis. Se utilizada uma UCP NX3020 ou NX3030 pode-se configurar pares NIC Teaming, utilizando no máximo o número de módulos suportados por cada

UCP, como por exemplo na arquitetura mostrada na Figura 4-9, onde temos um par NIC Teaming e mais uma interface Ethernet independente, utilizando três módulos.

Para agrupar dois módulos NX5000 como um par redundante, estes dois módulos devem necessariamente ocupar posições vizinhas no bastidor, e o checkbox "Redundância de Comunicação" do módulo à esquerda deve ser marcado, como mostra a Figura 4-10.

Ao fazer isto, o módulo à direita tem a edição de seus parâmetros bloqueada. Os parâmetros editados no módulo à esquerda passam a ser comuns para os dois módulos.

Por outro lado, desmarcando o checkbox "Redundância de Comunicação" do módulo à esquerda, provoca-se a separação dos módulos, que voltam a se comportar como módulos individuais sem redundância NIC Teaming.

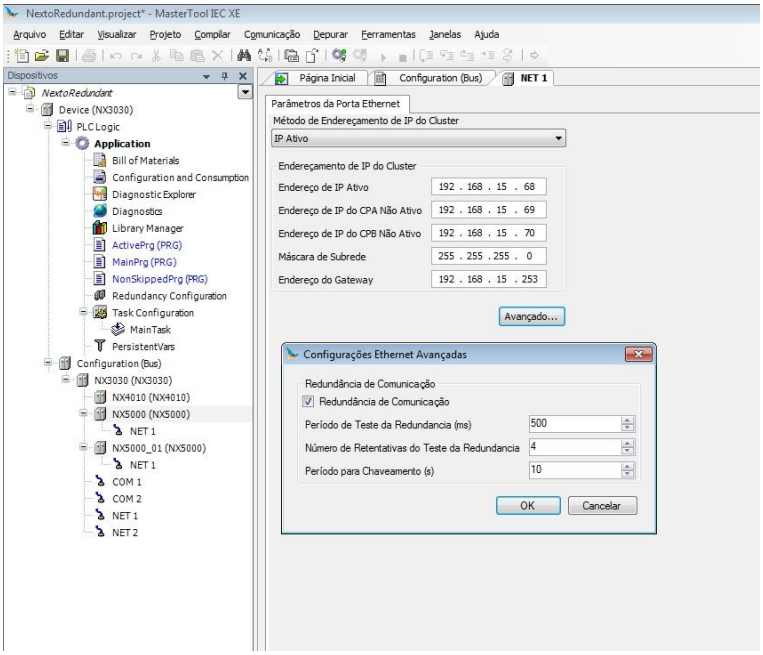

Figura 4-10. Parâmetro de Redundância NX5000

Configuração de Protocolos

Independente dos protocolos utilizados em cada aplicação, as UCPs da Série Nexto possuem alguns limites máximos de utilização para cada modelo de UCP. Existem basicamente dois diferentes tipos de protocolos de comunicação, os que utilizam mapeamentos simbólicos e os utilizam mapeamentos por representação direta. O limite máximo de mapeamentos assim como a quantidade máxima de protocolos (instâncias) está definida na Tabela 4-17:

	NX3004	NX3010	NX3020	NX3030
Pontos mapeados	20480	20480	20480	20480
Mapeamentos para MODBUS simbólico	5120	5120	5120	5120
Mapeamentos/requisições MODBUS (representação direta/simbólico respectivamente)	512	512	512	512
NETs – Instâncias Cliente ou Servidor	4	4	8	16
COM (n) – Instâncias Mestre ou Escravo	1	1	1	1

Tabela 4-17. Limites dos Protocolos por UCP

Notas:

Pontos mapeados: Cada variável ou item de um tipo de dado será considerado um ponto. O mesmo é considerado para cada posição do tipo ARRAY. Isso significa que a quantidade de pontos mapeados é incrementada em uma unidade quando houver uma variável de tipo simples sendo declarada, e no caso de uma variável do tipo ARRAY, será incrementada de acordo com o tamanho do ARRAY declarado. Isto é independente do tamanho do tipo de dado, então mapear uma variável do tipo INT (16 bits) em um Holding Register dos drivers MODBUS simbólicos ou uma variável do tipo LINT (64 bits) em quatro Holding Registres dos drivers MODBUS simbólicos é contabilizado como apenas um ponto mapeado.

Mapeamentos para MODBUS simbólico: Um mapeamento é uma relação entre uma variável interna de aplicação e um objeto do protocolo de aplicação. O valor limite para mapeamentos do projeto corresponde a soma de todos os mapeamentos realizados nas instâncias de protocolos de comunicação e seus respectivos dispositivos.

Mapeamentos/requisições MODBUS (representação direta/simbólico respectivamente): Limite de mapeamentos (linhas) para o MODBUS por representação direta e de requisições para o MODBUS por mapeamento simbólico.

ATENÇÃO:
Nos casos em que os dois tipos de protocolos forem usados em conjunto, à medida que um tipo é utilizado a capacidade máxima do outro tipo diminui. Por exemplo, se forem utilizados 10240 mapeamentos simbólicos só será possível utilizar 256 mapeamentos por representação direta. A razão entre os dois tipos de mapeamentos é de 40 mapeamentos simbólicos para cada mapeamento por representação direta.

NETs: Instâncias Cliente ou Servidor: O valor máximo definido acima é distribuído entre todas as interfaces Ethernet do sistema, ou seja, inclui os módulos de expansão, quando estes são aplicados. Exemplos para esse tipo de tarefa são as instâncias do Protocolo MODBUS.

COM (n): Instâncias Mestre ou Escravo: O "n" representa o número da interface serial, ou seja, mesmo com módulos de expansão, o valor da tabela será o máximo por interface. Exemplos para esse tipo de tarefa são as instâncias do Protocolo MODBUS.

O número máximo de instâncias são concorrentes entre si, ou seja, entre o MODBUS RTU Mestre e Escravo somente uma instância pode ser configurada por interface em qualquer modelo de UCP. Entre MODBUS Ethernet Cliente e Servidor somente quatro (NX3010), oito (NX3020) ou dezesseis (NX3030) instâncias podem ser configuradas por projeto.

As limitações do protocolo MODBUS por Representação Direta e por mapeamentos simbólicos para as UCPs podem ser visualizadas na Tabela 4-18 e na Tabela 4-19, respectivamente.

Limitações	MODBUS RTU Mestre	MODBUS RTU Escravo	MODBUS Ethernet Cliente	MODBUS Ethernet Servidor
Mapeamentos por instância	128	32	128	32
Dispositivos por instância	64	1(1)	64	64(2)
Mapeamentos por dispositivo	32	32	32	32
Requisições simultâneas por instância	-	-	128	64
Requisições simultâneas por dispositivo	-	-	8	64

Tabela 4-18. Limites do Protocolo MODBUS por Representação Direta

Notas:

Dispositivos por instância:

- **Protocolos Mestre ou Cliente:** quantidade de dispositivos escravos ou servidores suportados por cada instância de protocolo Mestre ou Cliente.
- **Protocolo MODBUS RTU Escravo:** o limite (1) informado diz respeito às interfaces seriais, que não permitem que um Escravo possa estabelecer comunicação, simultaneamente, através da mesma interface serial, com mais de um dispositivo Mestre. Não é necessário, e nem é possível, declarar nem configurar o dispositivo Mestre sob a instância do protocolo MODBUS RTU Escravo. O dispositivo Mestre terá acesso a todos os mapeamentos feitos diretamente na instância do protocolo MODBUS RTU Escravo.
- **Protocolo MODBUS Servidor:** o limite (2) informado diz respeito às interfaces Ethernet, que limitam a quantidade de conexões que podem ser estabelecidas com outros dispositivos através de uma mesma interface Ethernet. Não é necessário, e nem é possível, declarar nem configurar os dispositivos Clientes sob a instância do protocolo MODBUS Servidor. Todos os dispositivos Clientes terão acesso a todos os mapeamentos feitos diretamente na instância do protocolo MODBUS Servidor.

Mapeamentos por dispositivo: O número máximo de mapeamentos por dispositivo apesar de estar relacionado acima, também é limitado pelo número máximo de mapeamentos do protocolo. Também deve ser considerado o limite máximo de mapeamentos da UCP conforme na Tabela 4-17.

Requisições simultâneas por instância: Quantidade de requisições que podem ser transmitidas simultaneamente por cada instância de protocolo Cliente, ou que podem ser recebidas simultaneamente por cada instância de protocolo Servidor. Instâncias de protocolo MODBUS RTU, Mestre ou Escravo, não suportam requisições simultâneas.

Requisições simultâneas por dispositivo: Quantidade de requisições que podem ser transmitidas simultaneamente para cada dispositivo MODBUS Servidor, ou que podem ser recebidas simultaneamente de cada dispositivo MODBUS Cliente. Dispositivos MODBUS RTU, Mestre ou Escravo não suportam requisições simultâneas.

Limitações	MODBUS RTU Mestre	MODBUS RTU Escravo	MODBUS Ethernet Cliente	MODBUS Ethernet Servidor
Dispositivos por instância	64	1(1)	64	64(2)
Requisições por dispositivo	32	-	32	-
Requisições simultâneas por instância	-	-	128	64
Requisições simultâneas por dispositivo	-	-	8	64

Tabela 4-19. Limites do Protocolo MODBUS por Mapeamentos Simbólicos

Notas:

Dispositivos por instância:

- **Protocolos Mestre ou Cliente:** quantidade de dispositivos escravos ou servidores suportados por cada instância de protocolo Mestre ou Cliente.
- **Protocolo MODBUS RTU Escravo:** o limite (1) informado diz respeito às interfaces seriais, que não permitem que um Escravo possa estabelecer comunicação, simultaneamente, através da mesma interface serial, com mais de um dispositivo Mestre. Não é necessário, e nem é possível, declarar nem configurar o dispositivo Mestre sob a instância do protocolo MODBUS RTU Escravo. O dispositivo Mestre terá acesso a todos os mapeamentos feitos diretamente na instância do protocolo MODBUS RTU Escravo.

- **Protocolo MODBUS Servidor:** o limite [(2)] informado diz respeito às interfaces Ethernet, que limitam a quantidade de conexões que podem ser estabelecidas com outros dispositivos através de uma mesma interface Ethernet. Não é necessário, e nem é possível, declarar nem configurar os dispositivos Clientes sob a instância do protocolo MODBUS Servidor. Todos os dispositivos Clientes terão acesso a todos os mapeamentos feitos diretamente na instância do protocolo MODBUS Servidor.

Requisições por dispositivo: Quantidade de requisições, como por exemplo de leitura ou escrita de holding registers, que podem ser configuradas para cada um dos dispositivos (escravos ou servidores) de instâncias de protocolos Mestre ou Cliente. Este parâmetro não é aplicável às instâncias de protocolos Escravo ou Servidor.

Requisições simultâneas por instância: Quantidade de requisições que podem ser transmitidas simultaneamente por cada instância de protocolo Cliente, ou que podem ser recebidas simultaneamente por cada instância de protocolo Servidor. Instâncias de protocolo MODBUS RTU, Mestre ou Escravo, não suportam requisições simultâneas.

Requisições simultâneas por dispositivo: Quantidade de requisições que podem ser transmitidas simultaneamente para cada dispositivo MODBUS Servidor, ou que podem ser recebidas simultaneamente de cada dispositivo MODBUS Cliente. Dispositivos MODBUS RTU, Mestre ou Escravo não suportam requisições simultâneas.

ATENÇÃO: Os drivers de comunicação por Mapeamentos simbólicos estão disponíveis apenas a partir da versão 1.3.0.20 das UCPs da Série Nexto. Da mesma forma para utilizar esta funcionalidade é necessária à versão 1.40 ou superior do MasterTool IEC XE.

Comportamento dos Protocolos x Estados da UCP

A Tabela 4-20 mostra detalhadamente cada comportamento assumido para cada protocolo de comunicação configurável nas UCPs da série Nexto, em todos seus estados de funcionamento.

Protocolo	Tipo	Estado operacional da UCP					
		STOP			RUN		
		Depois do download antes da aplicação iniciar	Depois de a aplicação ir para STOP (PAUSE)	Depois de uma exceção	Não redundante ou Ativo	Redundante em Reserva	Após um breakpoint na MainPrg
MODBUS	Slave/Server	×	×	×	✓	✓	×
	Master/Client	×	×	×	✓	✓	✓
SOE (DNP3)	Outstation	✓	✓	✓	✓	×	✓
EtherCAT	Master	×	×	×	✓	NA	×
OPC	Server	✓	✓	✓	✓	×	✓
SNTP	Client	✓	✓	✓	✓	✓	✓
HTTP	Server	✓	✓	✓	✓	✓	✓
SNMP	Agent	✓	✓	✓	✓	✓	✓

Tabela 4-20. Comportamento dos Protocolos x Estados da UCP

Notas:

Símbolo ✓: O protocolo permanece ativo e operando normalmente.

Símbolo ×:O protocolo está desabilitado.

EtherCAT: Os testes foram executados usando a *MainTask* como *Tarefa de Barramento EtherCAT*, caso outra tarefa seja utilizada, o protocolo permanecerá ativo quando ocorrer um breakpoint na *MainPrg*. O protocolo EtherCAT não está disponível para os modelos NX3004 e NX3010 e nem para configuração de redundância de cluster.

SOE: O protocolo de registro de eventos (SOE) não está disponível para os modelos NX3004 e NX3010.

MODBUS RTU Mestre

Este protocolo está disponível para as UCPs da Série Nexto nos seus canais seriais. Ao selecionar esta opção no MasterTool IEC XE, a UCP passa a ser mestre da comunicação MODBUS, possibilitando o acesso a outros dispositivos com o mesmo protocolo, quando esta estiver em modo de execução (Modo Run).

Há dois modos de configuração para este protocolo. Um deles faz uso de Representação Direta (%Q), no qual as variáveis são definidas pelo seu endereço. O outro, chamado Mapeamento Simbólico, tem as variáveis definidas pelo seu nome.

Independente do modo de configuração, os passos para inserir uma instância do protocolo e configurar a interface serial são iguais. O procedimento para inserir uma instância de protocolo é encontrado com detalhes no Manual de Utilização do MasterTool IEC XE – MU299048 ou no capítulo Inserindo uma Instância de Protocolo. As demais etapas de configuração serão descritas a seguir para cada modo.

- Adicionar a instância do protocolo MODBUS RTU Mestre ao canal serial COM 1 ou COM 2 (ou ambos, em caso de duas redes de comunicação). Para realizar esse procedimento, consultar o capítulo Programação Inicial – Inserindo uma Instância de Protocolo.
- Configurar a interface serial, escolhendo a taxa de transmissão, o comportamento dos sinais de modem RTS/CTS, a paridade, bits de parada do canal e demais configurações através de um duplo clique sobre o canal serial COM 1 ou COM 2.
- Consultar o capítulo Configuração - Configuração das Interfaces Seriais.

Configuração do Protocolo MODBUS Mestre por Mapeamento Simbólico

Para configurar este protocolo usando Mapeamento Simbólico, é necessário executar os seguintes passos:

- Configurar os parâmetros gerais do protocolo MODBUS Mestre, como: tempos de atraso de envio e interframe mínimo como na Figura 4-11.
- Adicionar e configurar dispositivos através da aba Parâmetros Gerais, definindo endereço do escravo, time-out de comunicação e número de retentativas de comunicação como pode ser visto na Figura 4-12.
- Adicionar e configurar os mapeamentos MODBUS na aba Mapeamentos conforme Figura 4-13, especificando o nome da variável, tipo de dados e endereço inicial do dado, o tamanho do dado e a faixa são preenchidos automaticamente.
- Adicionar e configurar as requisições MODBUS como apresentado na Figura 4-14, especificando a função, o tempo de varredura da requisição, o endereço inicial (leitura/escrita), o tamanho dos dados (Leitura/Escrita) e gerar as variáveis de diagnóstico e desabilitação das requisições através dos botões na parte inferior da janela.

Parâmetros Gerais do Protocolo MODBUS Mestre – Configuração por Mapeamento Simbólico

Os parâmetros gerais, encontrados na tela inicial de configuração do protocolo MODBUS (Figura 4-11), são definidos como:

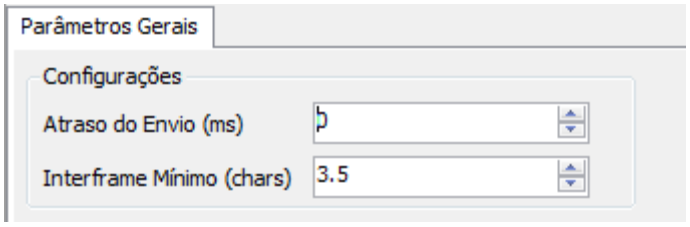

Figura 4-11. Tela de Configuração Parâmetros Gerais MODBUS RTU Mestre

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Atraso do Envio (ms)	Tempo de atraso para envio da resposta.	0	0 a 65535
Interframe Mínimo (chars)	Tempo mínimo de silêncio entre diferentes frames.	3.5	3.5 a 100.0

Tabela 4-21. Configurações Gerais MODBUS RTU Mestre

Notas:

Atraso do envio: A resposta à uma requisição MODBUS pode causar problemas em certos momentos, como, por exemplo, na interface RS-485 ou outra half-duplex. Às vezes existe um atraso entre o tempo de resposta do escravo e o silêncio na linha física (atraso no escravo para colocar RTS em zero e colocar o transmissor RS-485 em alta impedância). Para resolver o problema, o mestre pode esperar o tempo determinado nesse campo antes de enviar a nova requisição. Caso contrário, os primeiros bytes transmitidos pelo mestre, durante a requisição, podem ser perdidos.

Interframe mínimo: A norma MODBUS define esse tempo como 3,5 caracteres, porém esse parâmetro é configurável para atender aos dispositivos que não estão de acordo com o padrão.

Os diagnósticos e comandos do protocolo MODBUS Mestre configurado, seja por mapeamento simbólico ou por representação direta, são armazenados em variáveis do tipo T_DIAG_MODBUS_RTU_MASTER_1 e ainda para o mapeamento por representação direta estão em 4 bytes e 8 words, os quais estão descritas na Tabela 4-22 (n é o valor configurado no campo Endereço Inicial de Diagnósticos em %Q):

Variável de Representação Direta	Variável de diagnóstico do tipo T_DIAG_MODBUS_RTU_MASTER_1.*		Tamanho	Descrição
Bits de diagnóstico:				
%QX(n).0	tDiag.*	bRunning	BIT	O mestre está em execução.
%QX(n).1		bNotRunning	BIT	O mestre não está em execução (ver bit: bInterruptedByCommand).
%QX(n).2		bInterruptedByCommand	BIT	O bit bNotRunning foi habilitado, pois o mestre foi interrompido pelo usuário através de bits de comando.
%QX(n).3		bConfigFailure	BIT	Diagnóstico descontinuado.
%QX(n).4		bRXFailure	BIT	Diagnóstico descontinuado.
%QX(n).5		bTXFailure	BIT	Diagnóstico descontinuado.
%QX(n).6		bModuleFailure	BIT	Indica se há falha no módulo ou o módulo não está presente
%QX(n).7		bDiag_7_reserved	BIT	Reservado
Códigos de erro:				
%QB(n+1)	eErrorCode		SERIAL_STATUS (BYTE)	0: não existem erros. 1: porta serial inválida. 2 :modo da porta serial inválido 3: baud rate inválido. 4: data bits inválido. 5: paridade inválida. 6: stop bits inválido. 7: parâmetro de sinal de modem inválido. 8: parâmetro de Threshold de RX da UART inválido. 9: parâmetro de time-out inválido. 10: porta serial ocupada. 11: erro de hardware na UART. 12: erro de hardware remoto. 20: tamanho do buffer de transmissão inválido. 21: método de sinal de modem inválido.

				22: time-out de CTS = verdadeiro. 23: time-out de CTS = falso. 24: erro de time-out na transmissão. 30: tamanho do buffer de recepção inválido. 31: erro de time-out na recepção. 32: controle de fluxo configurado diferente de manual. 33: controle de fluxo inválido para a porta serial configurada. 34: recepção de dados não permitida no modo normal. 35: recepção de dados não permitida no modo estendido. 36: interrupção DCD não permitida. 37: interrupção CTS não permitida. 38: interrupção DSR não permitida. 39: porta serial não configurada. 50: erro interno na porta serial
Bits de comando, reiniciados automaticamente:				
%QX(n+2).0	tCommand.*	bStop	BIT	Parar o mestre.
%QX(n+2).1		bRestart	BIT	Reiniciar o mestre.
%QX(n+2).2		bResetCounter	BIT	Reiniciar as estatísticas dos diagnósticos (contadores).
%QX(n+2).3		bDiag_19_reserved	BIT	Reservado
%QX(n+2).4		bDiag_20_reserved	BIT	Reservado
%QX(n+2).5		bDiag_21_reserved	BIT	Reservado
%QX(n+2).6		bDiag_22_reserved	BIT	Reservado
%QX(n+2).7		bDiag_23_reserved	BIT	Reservado
%QB(n+3)	byDiag_03_reserved		BYTE	Reservado
Estatísticas de comunicação:				
%QW(n+4)	tStat.*	wTXRequests	WORD	Contador de requisições transmitidas pelo mestre (0 a 65535).
%QW(n+6)		wRXNormalResponses	WORD	Contador de respostas normais recebidas pelo mestre (0 a 65535).
%QW(n+8)		wRXExceptionResponses	WORD	Contador de respostas com códigos de exceção recebidas pelo mestre (0 a 65535).
%QW(n+10)		wRXIllegalResponses	WORD	Contador de respostas ilegais recebidas pelo mestre – sintaxe inválida, número insuficiente de bytes recebidos, CRC inválido – (0 a 65535).
%QW(n+12)		wRXOverrunErrors	WORD	Contador de erros de overrun durante a recepção – UART FIFO ou fila RX – (0 a 65535).
%QW(n+14)		wRXIncompleteFrames	WORD	Contador de respostas com erro de construção, paridade ou falha durante a recepção (0 a 65535).
%QW(n+16)		wCTSTimeOutErrors	WORD	Contador de erro de time-out no CTS, utilizando o handshake RTS/CTS, durante a transmissão (0 a 65535).
%QW(n+18)			WORD	Reservado

Tabela 4-22. Diagnósticos MODBUS RTU Mestre

Nota:

Contadores: Todos os contadores dos diagnósticos do MODBUS RTU Mestre retornam à zero quando o valor limite 65535 é ultrapassado.

Configuração dos Dispositivos – configuração por Mapeamento Simbólico

A configuração dos dispositivos escravos, visualizados na Figura 4-12, seguem os seguintes parâmetros:

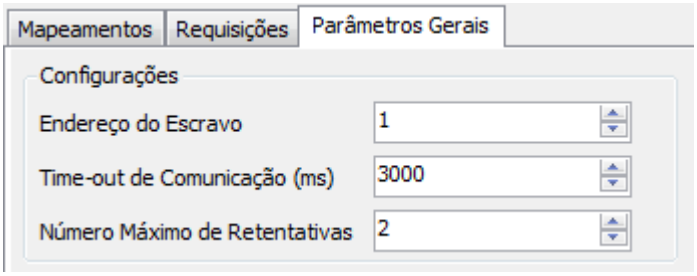

Figura 4-12. Tela de Configurações dos Parâmetros Gerais do Dispositivo

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Endereço do escravo	Endereço do escravo MODBUS	1	0 a 255
Time-out de comunicação (ms)	Define o time-out do nível de aplicação	3000	10 a 65535
Número Máximo de Retentativas	Define o número de tentativas antes de reportar um erro de comunicação	2	0 a 9

Tabela 4-23. Configurações do Dispositivo

Notas:

Endereço do escravo: De acordo com a Norma MODBUS, a faixa de endereços válidos para escravos é de 0 a 247, sendo os endereços 248 a 255 reservados. Quando o mestre envia um comando de escrita com o endereço configurado como zero, ele está realizando requisições broadcast na rede.

Time-out de comunicação: O time-out da comunicação é o tempo que o mestre aguardará por uma resposta do escravo à requisição. Para um dispositivo MODBUS RTU Mestre devem ser levadas em consideração ao menos as seguintes variáveis do sistema: o tempo que o escravo leva para transmitir o frame (de acordo com a taxa de transmissão), o tempo que o escravo leva para processar a requisição e o delay de envio da resposta caso seja configurado no escravo. É recomendado que o time-out seja igual ou maior que o tempo para transmitir o frame somado ao delay de envio da resposta e a duas vezes o tempo de processamento da requisição. Para mais informações, ver capítulo Configuração - Configuração de Protocolos - Desempenho de Comunicação.

Número máximo de retentativas: Define o número de retentativas antes de reportar um erro de comunicação. Por exemplo, se o escravo não responder a um pedido e o mestre estiver configurado para enviar três retentativas, o número do contador de erros será incrementado por 1 unidade ao final da execução destas três retentativas. Após o incremento do erro o processo de tentativa de comunicação é reiniciado e caso o número de retentativas seja atingido novamente, novo erro será incrementado no contador.

Configuração dos Mapeamentos – Configuração por Mapeamento Simbólico

A configuração dos mapeamentos MODBUS, visualizadas na Figura 4-13, segue os parâmetros descritos na Tabela 4-24:

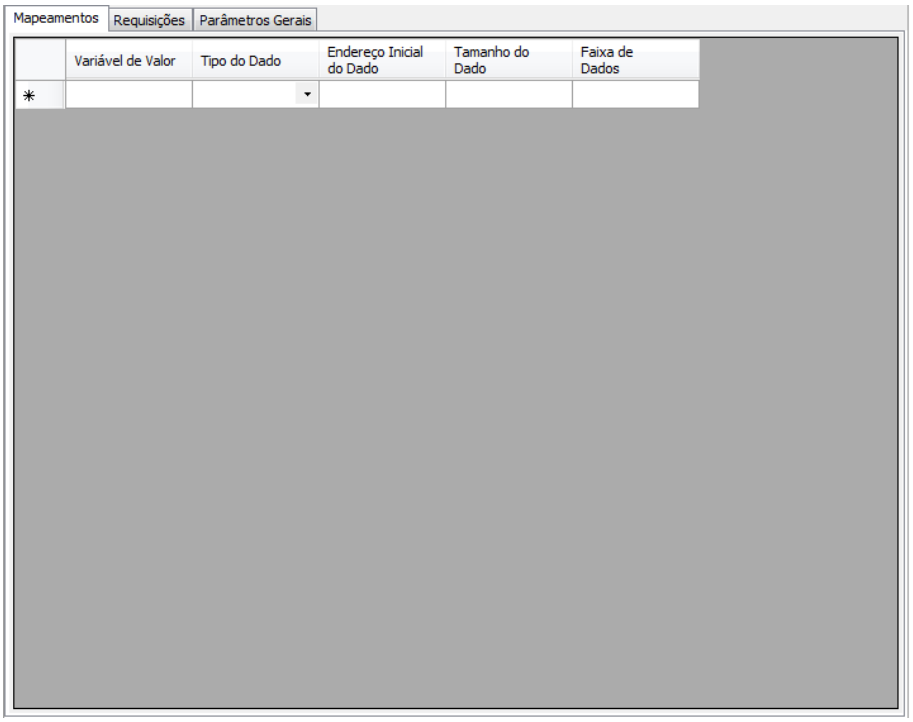

Figura 4-13. Tela de Mapeamentos de Dados MODBUS

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Variável de Valor	Nome da variável simbólica	-	Nome de uma variável declarada em um programa ou GVL
Tipo de Dado	Tipo de dado MODBUS	-	Coil Escrita(1 bit) Coil Leitura (1 bit) Holding Register Escrita (16 bits) Holding Register Leitura (16 bits) Holding Register – Máscara AND (16 bits) Holding Register – Máscara OR (16 bits) Input Register (16 bits) Input Status (1 bit)
Endereço Inicial do Dado	Endereço inicial dos dados MODBUS	-	1 a 65536
Tamanho do Dado	Tamanho do dado MODBUS	-	1 a 65536
Faixa de Dados	Faixa de endereços do dado configurado	-	-

Tabela 4-24. Configuração dos Mapeamentos MODBUS

Notas:

Variável de Valor: Esse campo é utilizado para especificar uma variável simbólica na relação MODBUS.

Tipo de Dado: Esse campo é utilizado para especificar o tipo de dado utilizado na relação MODBUS.

Tipo de Dado	Tamanho [bits]	Descrição
Coil Escrita	1	Saída digital de escrita.
Coil Leitura	1	Saída digital de leitura.
Holding Register de Escrita	16	Saída analógica de escrita.
Holding Register de Leitura	16	Saída analógica de leitura.

Holding Register com mascara AND	16	Saída analógica que pode ser lida ou escrita com máscara AND.
Holding Register com mascara OR	16	Saída analógica que pode ser lida ou escrita com máscara OR.
Input Register	16	Entrada analógica que pode ser apenas lida.
Input Status	1	Entrada digital que pode ser apenas lida.

Tabela 4-25. Tipos de Dados suportados no MODBUS RTU Mestre

Endereço Inicial do Dado: Endereço inicial do dado de um mapeamento MODBUS.

Tamanho do Dado: O valor de Tamanho especifica a quantidade máxima de dados que uma relação MODBUS poderá acessar, a partir do endereço inicial. Sendo assim, para ler uma faixa de endereços contínua, é necessário que todos os endereços estejam declarados em uma única relação. Este campo varia de acordo com o tipo de dado MODBUS configurado.

Faixa de Dados: Este campo mostra ao usuário a faixa de endereços de memória utilizada pela relação MODBUS.

Configuração das Requisições – configuração por Mapeamento Simbólico

A configuração das requisições MODBUS, visualizadas na Figura 4-14, segue os parâmetros descritos na Tabela 4-26:

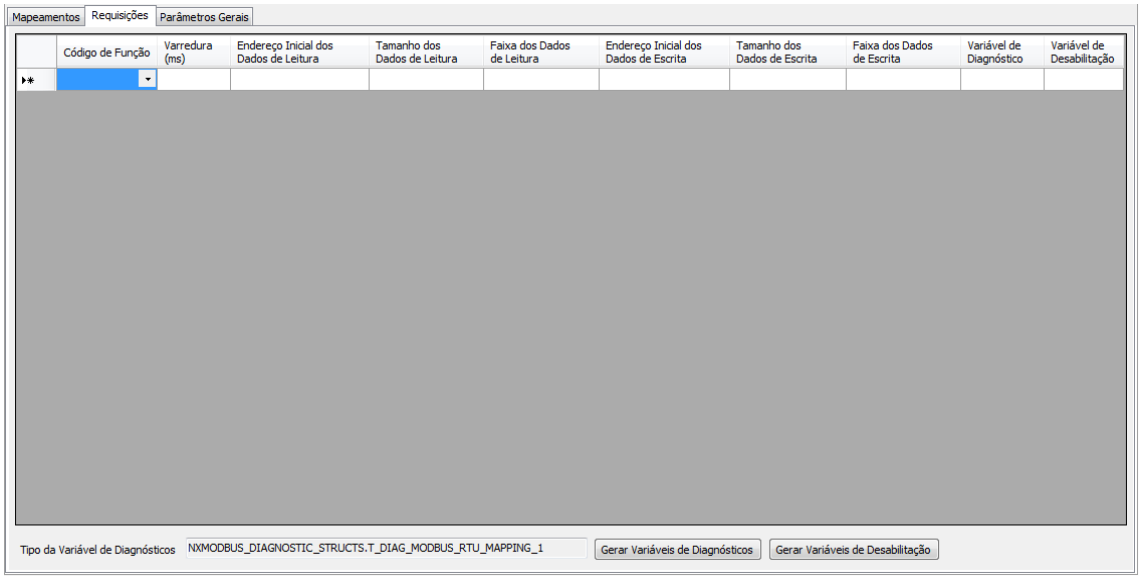

Figura 4-14. Tela de Requisições de dados MODBUS Mestre

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Código de Função	Tipo de função MODBUS.	-	01 – Leitura de Coils 02 – Leitura de Input Status 03 – Leitura de Holding Registers 04 – Leitura de Input Registers 05 – Escrita de um Coil 06 – Escrita de um Register 15 – Escrita de múltiplos Coils 16 – Escrita de Múltiplos Registers 22 –Escrita Mascarada de Register 23 – Leitura/Escrita de Múltiplos Register
Varredura (ms)	Período de comunicação (ms).	100	0 a 3600000
Endereço Inicial dos Dados de	Endereço inicial dos dados de leitura MODBUS.	-	1 a 65536

Leitura			
Tamanho dos Dados de Leitura	Tamanho dos dados de leitura MODBUS.	-	Depende da função utilizada
Faixa dos dados de leitura	Faixa de endereço dos dados de leitura MODBUS.	-	0 a 2147483646
Endereço Inicial dos Dados de Escrita	Endereço inicial dos dados de escrita MODBUS.	-	1 a 65536
Tamanho dos Dados de Escrita	Tamanho dos dados de escrita MODBUS.	-	Depende da função utilizada
Faixa dos dados de Escrita	Faixa de endereço dos dados de escrita MODBUS.	-	0 a 2147483647
Variável de Diagnóstico	Nome da variável de diagnóstico.	-	Nome de uma variável declarada em um programa ou GVL
Variável de Desabilitação	Variável utilizada para desabilitar a relação MODBUS.	-	Campo destinado a variável simbólica utilizada para desabilitar, individualmente, as requisições MODBUS configuradas. Esta variável deve ser do tipo BOOL. A variável pode ser simples ou elemento de array e pode estar em estruturas.

Tabela 4-26. Configuração das Relações MODBUS Mestre

Notas:

Configuração: O número de configurações, padrão de fábrica e os valores da coluna possibilidades, podem variar de acordo com o tipo de dado e função MODBUS (FC).

Código de Função: As funções MODBUS (FC) disponíveis são os seguintes:

Tipo de Função	Código		Descrição
	DEC	HEX	
Acesso a Variáveis	1	0x01	Leitura de coils (FC 01)
	2	0x02	Leitura de input status (FC 02)
	3	0x03	Leitura de holding registers (FC 03)
	4	0x04	Leitura de input registers (FC 04)
	5	0x05	Escrita de um coil (FC 05)
	6	0x06	Escrita de um holding register (FC 06)
	15	0x0F	Escrita de múltiplos coils (FC 15)
	16	0x10	Escrita de holding registers (FC 16)
	22	0x16	Máscara de escrita do register (FC 22)
	23	0x17	Leitura/escrita holding registers (FC 23)

Tabela 4-27. Funções MODBUS Suportadas pelas UCPs Nexto

Varredura: Este parâmetro indica com que frequência à comunicação definida por esta requisição deve ser executada. Ao ser finalizada uma comunicação será aguardado um tempo igual ao configurado no campo varredura e, após, será executada uma nova comunicação.

Endereço Inicial dos Dados de Leitura: Campo destinado ao endereço inicial dos dados de leitura MODBUS.

Tamanho dos Dados de Leitura: O valor mínimo para o tamanho dos dados de leitura é 1 e o valor máximo depende da função MODBUS (FC) utilizada, conforme abaixo:

- Leitura de Coils (FC 1): 2000
- Leitura de Input Status (FC 2): 2000
- Leitura de Holding Registers (FC 3): 125
- Leitura de Input Registers (FC 4): 125
- Leitura/Escrita de Holding Registers (FC 23): 121

Faixa dos Dados de Leitura: Este campo mostra a faixa de dados de leitura MODBUS configurada para cada requisição. O endereço inicial de leitura, somado ao tamanho do dado de leitura resultará na faixa de dados de leitura de cada uma das requisições.

Endereço Inicial dos Dados de Escrita: Campo destinado ao endereço inicial dos dados de escrita MODBUS.

Tamanho dos Dados de Escrita: O valor mínimo para o tamanho dos dados de escrita é 1 e o valor máximo depende da função MODBUS (FC) utilizada, conforme abaixo:

- Escrita de Um Coil (FC 5): 1
- Escrita de Um Holding Registers (FC 6): 1
- Escrita de Múltiplos Coils (FC 15): 1968
- Escrita de Holding Registers (FC 16): 123
- Máscara de Escrita do Register (FC 22): 1
- Leitura/Escrita de Holding Registers (FC 23): 121

Faixa dos Dados de Escrita: Este campo mostra a faixa de dados de escrita MODBUS configurada para cada requisição. O endereço inicial de escrita, somado ao tamanho do dado de escrita resultará na faixa de dados de escrita de cada uma das requisições.

Variável de Diagnóstico: Os diagnósticos da requisição MODBUS configurada, seja por mapeamento simbólico ou por representação direta , são armazenados em variáveis do tipo T_DIAG_MODBUS_RTU_MAPPING_1 e ainda para o mapeamento por representação direta estão em 4 bytes e 2 words, os quais estão descritos na Tabela 4-28 (n é o valor configurado no campo Endereço Inicial de Diagnósticos em %Q).

Variável de Representação Direta	Variável de diagnóstico do tipo T_DIAG_MODBUS_RTU_MAPPING_1.*		Tamanho	Descrição
Bits de estado da comunicação:				
%QX(n).0	byStatus.*	bCommIdle	BIT	Comunicação inativa (aguardando ser constada).
%QX(n).1		bCommExecuting	BIT	Comunicação ativa.
%QX(n).2		bCommPostponed	BIT	Comunicação adiada, pois o número máximo de requisições simultâneas foi atingido. As comunicações adiadas serão executadas na mesma sequência em que elas foram requisitadas, evitando a indeterminação. O tempo gasto nesse estado não é contabilizado para efeitos de time-out. Os bits bCommIdle e bCommExecuting são falsos quando o bit bCommPostponed é verdadeiro.
%QX(n).3		bCommDisabled	BIT	Comunicação desabilitada. O bit bCommIdle é reiniciado nessa condição.
%QX(n).4		bCommOk	BIT	Comunicação finalizada anteriormente foi realizada com sucesso.
%QX(n).5		bCommError	BIT	Comunicação finalizada anteriormente teve um erro. Verificar código de erro.
%QX(n).6		bCommAborted	BIT	Não utilizado no MODBUS Mestre RTU.
%QX(n).7		bDiag_7_reserved	BIT	Reservado
Último código de erro (habilitado quando o bCommError = verdadeiro):				
%QB(n+1)	eLastErrorCode		MASTER_ERROR_CODE (BYTE)	Informa a possível causa do último erro ocorrido na relação MODBUS. Consulte a Tabela

			4-29 para detalhes das possibilidades.
Último código de exceção recebido pelo mestre:			
%QB(n+2)	eLastExceptionCode	MODBUS_EXCEPTION (BYTE)	NO_EXCEPTION (0) FUNCTION_NOT_SUPPORTED (1) MAPPING_NOT_FOUND (2) ILLEGAL_VALUE (3) ACCESS_DENIED (128)* MAPPING_DISABLED (129)* IGNORE_FRAME (255)*
Estatísticas de comunicação:			
%QB(n+3)	byDiag_3_reserved	BYTE	Reservado.
%QW(n+4)	wCommCounter	WORD	Contador de comunicações finalizadas, com ou sem erros. O usuário pode testar quando a comunicação foi finalizada testando a variação desse contador. Quando o valor 65535 é atingido, o contador retorna à zero.
%QW(n+6)	wCommErrorCounter	WORD	Contador de comunicações finalizadas com erros. Quando o valor 65535 é atingido, o contador retorna a zero.

Tabela 4-28. Diagnósticos das relações MODBUS

Códigos de exceção: Os códigos de exceção apresentados neste campo são os valores retornados pelo escravo. As definições dos códigos de exceção 128, 129 e 255, apresentadas nessa tabela, são válidas apenas na utilização de escravos Altus. Para escravos de outros fabricantes esses códigos de exceção podem ter significados diferentes.

Variável de Desabilitação: Campo destinado a variável do tipo booleana utilizada para desabilitar, individualmente, as requisições MODBUS configuradas na aba Requisições através do botão na parte inferior da janela. A requisição é desabilitada quando a variável, correspondente a requisição, for igual a 1, caso contrário, o a requisição está habilitada.

Último código de Erro: Os códigos das possíveis situações que ocasionam erro na comunicação MODBUS podem ser consultados abaixo:

Código	Enumerável	Descrição
1	ERR_EXCEPTION	Resposta reportada em um código de exceção (ver eLastExceptionCode = Código de Exceção).
2	ERR_CRC	Resposta com CRC inválido.
3	ERR_ADDRESS	Endereço MODBUS não encontrado. O endereço que respondeu à requisição foi diferente do esperado.
4	ERR_FUNCTION	Código inválido da função. A função recebida na resposta foi diferente da esperada pela requisição.
5	ERR_FRAME_DATA_COUNT	A quantidade de dados da resposta foi diferente do esperado
7	ERR_NOT_ECHO	Resposta não é eco da pergunta (FC 5 e 6).
8	ERR_REFERENCE_NUMBER	Número de referência inválido (FC 15 e 16).
9	ERR_INVALID_FRAME_SIZE	Resposta menor do que a esperada.
21	ERR_SEND	Erro durante a fase de transmissão.
22	ERR_RECEIVE	Erro durante a fase de recepção.
41	ERR_SEND_TIMEOUT	Time-out no nível de aplicação durante a transmissão.
42	ERR_RECEIVE_TIMEOUT	Time-out no nível de aplicação enquanto aguarda resposta.
43	ERR_CTS_OFF_TIMEOUT	Time-out enquanto aguarda CTS = falso na transmissão.
44	ERR_CTS_ON_TIMEOUT	Time-out enquanto aguarda CTS = verdadeiro na transmissão.
128	NO_ERROR	Sem erro desde a inicialização.

Tabela 4-29. Códigos de Erro das relações MODBUS Mestre

ATENÇÃO: Diferentemente de outras tarefas de uma aplicação, quando for atingida uma marca de depuração na MainTask, a tarefa de uma instância MODBUS RTU Mestre, e qualquer outra tarefa MODBUS, irá parar de ser executada no momento em que tentar efetuar uma escrita em uma área de memória. Isto ocorre para manter a consistência dos dados das áreas de memória enquanto a MainTask não estiver em execução.

Configuração do Protocolo MODBUS Mestre por Representação Direta (%Q)

Para configurar este protocolo usando representação direta (%Q), é necessário executar os seguintes passos:

- Configurar os parâmetros gerais do protocolo MODBUS Mestre, como: tempos de comunicação e variáveis de representação direta (%Q) para receber os diagnósticos.
- As descrições de cada configuração estão relacionadas a seguir, neste capítulo.
- Adicionar e configurar dispositivos, definindo endereço, variáveis de representação direta (%Q) para desabilitar as relações, timeouts e número de retentivas de comunicação.
- Adicionar e configurar relações MODBUS, especificando o tipo de dado e função MODBUS, timeouts, variáveis de representação direta (%Q) para receber os diagnósticos da relação e outras para receber/escrever os dados, quantidade de dados a comunicar e polling da relação.

As descrições de cada configuração estão relacionadas a seguir, neste capítulo.

Parâmetros Gerais do Protocolo MODBUS Mestre – configuração por Representação Direta (%Q)

Os parâmetros gerais, encontrados na tela inicial de configuração do protocolo MODBUS (Figura 4-15), são definidos como:

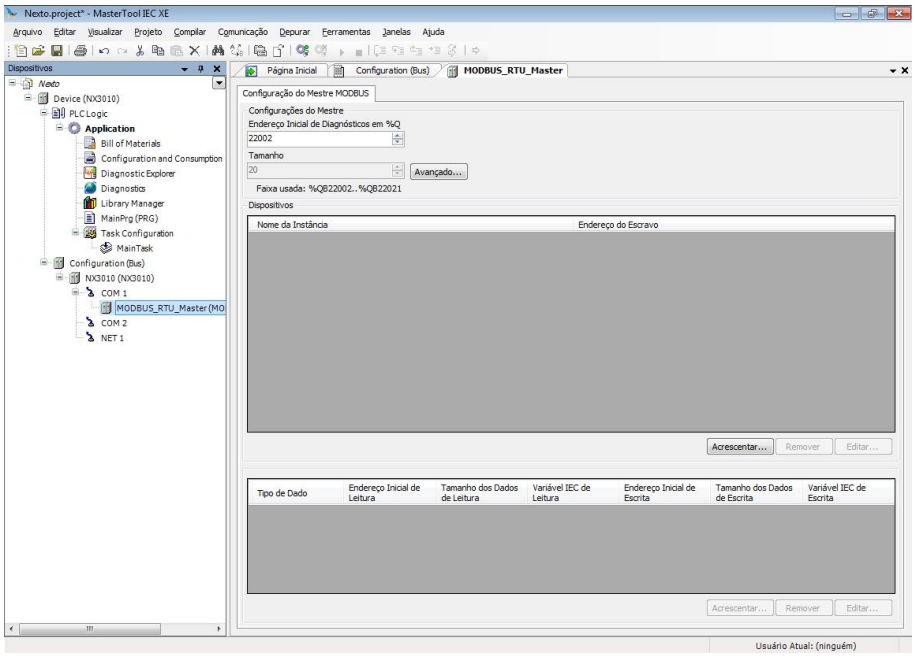

Figura 4-15. Tela de Configuração MODBUS RTU Mestre

Variáveis de representação direta (%Q) para os diagnósticos do protocolo:

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Endereço Inicial de Diagnósticos em %Q	Endereço inicial das variáveis de diagnóstico	-	0 a 2147483628
Tamanho	Tamanho da área de diagnósticos	20	Desabilitado para edição

Tabela 4-30. Configurações do Mestre – MODBUS Mestre

Notas:

Endereço Inicial de Diagnósticos em %Q: Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Padrão de Fábrica: O padrão de fábrica não pode ser definido para o campo Endereço Inicial de Diagnósticos em %Q, pois a criação de uma instância do protocolo pode ser realizada em qualquer momento no desenvolvimento do aplicativo, fazendo com que o próprio software MasterTool IEC XE aloque um valor, da faixa de variáveis de saída de representação direta (%Q), ainda não utilizado.

Os diagnósticos e comandos do protocolo MODBUS estão descritoas na Tabela 4-22.

Os tempos de comunicação do protocolo MODBUS Mestre, encontrados no botão "Avançado..." da tela de configuração, estão divididos em Atraso do Envio e Interframe Mínimo, maiores detalhes estão descritos na seção Parâmetros Gerais do Protocolo MODBUS Mestre – Configuração por Mapeamento Simbólico.

Configuração dos Dispositivos – Configuração por Representação Direta (%Q)

A configuração dos dispositivos escravos, visualizados na Figura 4-16, seguem os seguintes parâmetros:

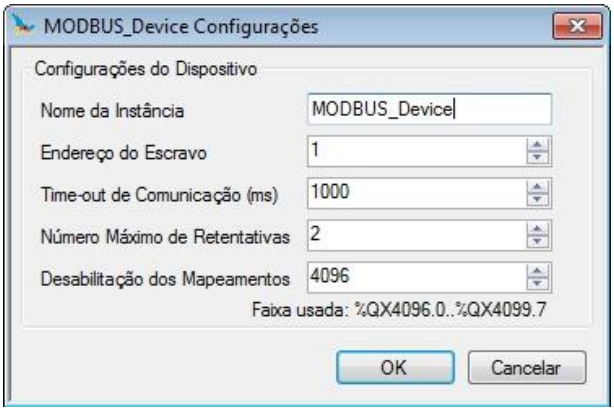

Figura 4-16. Configurações do Dispositivo

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Nome da Instância	Nome da instância	MODBUS_Device	Identificador, segundo a IEC 61131-3
Endereço do Escravo	Endereço do escravo MODBUS	1	0 a 255
Time-out de Comunicação (ms)	Define o time-out do nível de aplicação	1000	10 a 65535
Número Máximo de Retentativas	Define o número de tentativas antes de reportar um erro de comunicação	2	0 a 9
Desabilitação dos Mapeamentos	Endereço inicial utilizado para desabilitar as relações MODBUS	-	0 a 2147483644

Tabela 4-31. Configurações do Dispositivo – MODBUS Mestre

Notas:

Nome da Instância: Esse campo é o identificador do dispositivo, o qual é verificado segundo a IEC 61131-3, ou seja, não permite espaços, caracteres especiais e iniciar com caractere numeral. É limitado em 24 caracteres.

Desabilitação dos Mapeamentos: Composto por 32 bits, utilizados para desabilitar, individualmente, as 32 relações MODBUS configuradas no espaço Mapeamento do Dispositivo. A relação é desabilitada quando o bit, correspondente a relação, for igual a 1, caso contrário, o mapeamento está habilitado. Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Padrão de Fábrica: O padrão de fábrica não pode ser definido para o campo Desabilitação dos Mapeamentos, pois a criação de uma instância do protocolo pode ser realizada em qualquer momento no desenvolvimento do aplicativo, fazendo com que o próprio software MasterTool IEC XE aloque um valor, da faixa de variáveis de saída de representação direta (%Q), ainda não utilizado.

Para maiores detalhes dos parâmetros Endereço do Escravo, Time-out de Comunicação e Número máximo de retentativas ver notas na seção Configuração dos Dispositivos – configuração por Mapeamento Simbólico.

Configuração dos Mapeamentos – configuração por Representação Direta (%Q)

A configuração das relações MODBUS, visualizadas na Figura 4-17 e Figura 4-18 , segue os parâmetros descritos na Tabela 4-32:

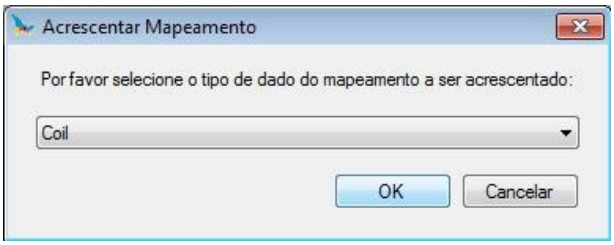

Figura 4-17. Tipo de Dado MODBUS

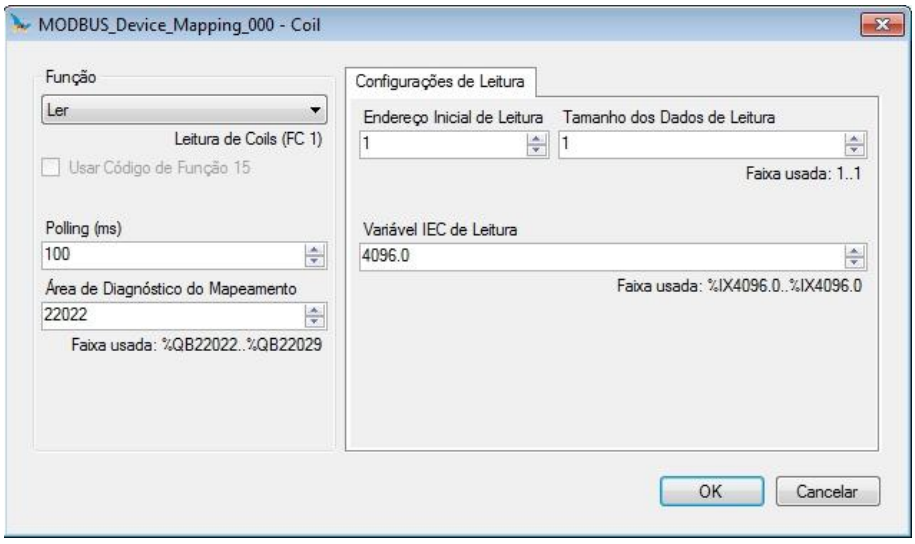

Figura 4-18. Função MODBUS

Na Tabela 4-32, o número de configurações, padrão de fábrica e os valores da coluna possibilidades, podem variar de acordo com o tipo de dado e função MODBUS (FC).

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Função	Tipo de função MODBUS.	Ler	Ler Escrever Ler/Escrever Máscara de

			Escrita
Polling (ms)	Período de comunicação (ms).	100	0 a 3600000
Área de Diagnóstico do Mapeamento	Endereço inicial dos diagnósticos da relação MODBUS (%Q).	-	0 a 2147483640
Endereço Inicial de Leitura	Endereço inicial dos dados de leitura MODBUS.	1	1 a 65536
Tamanho dos Dados de Leitura	Número de dados de leitura MODBUS.	-	Depende da função utilizada
Variável IEC de Leitura	Endereço inicial das variáveis de leitura (%I).	-	0 a 2147483646
Endereço Inicial de Escrita	Endereço inicial dos dados de escrita MODBUS.	1	1 a 65536
Tamanho dos Dados de Escrita	Número de dados de escrita MODBUS.	-	Depende da função utilizada
Variável IEC de Escrita	Endereço inicial das variáveis de escrita (%Q).	-	0 a 2147483647
Máscara de Escrita das Variáveis IEC	Endereço inicial das variáveis para a máscara de escrita (%Q).	-	0 a 2147483644

Tabela 4-32. Mapeamentos do Dispositivo

Notas:

Função: Os tipos de dados disponíveis estão detalhados na Tabela 4-25 e as funções MODBUS (FC) disponíveis na Tabela 4-27.

Polling: Este parâmetro indica com que frequência a comunicação definida por esta relação deve ser executada. Ao ser finalizada uma comunicação será aguardado um tempo igual ao polling configurado e, após, será executada uma nova comunicação o mais rápido possível.

Área de Diagnósticos do Mapeamento: Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas. Os diagnósticos da relação MODBUS configurada estão descritos na Tabela 4-28.

Tamanho dos Dados de Leitura e de Escrita: Detalhes do tamanho dos dados suportados por cada função está descrito nas notas da seção Configuração das Requisições – configuração por Mapeamento Simbólico.

Variável IEC de Leitura: Caso o tipo de dado MODBUS seja Coil ou Input Status (bit), o endereço inicial das variáveis IEC de leitura terá o formato por exemplo %IX10.1. Porém, se o tipo de dado MODBUS for Holding Register ou Input Register (16 bits), o endereço inicial das variáveis IEC de leitura terá o formato %IW. Esse campo é limitado pelo tamanho da memória de variáveis de entrada endereçáveis (%I) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Variável IEC de Escrita: Caso o tipo de dado MODBUS seja Coil, o endereço inicial das variáveis IEC de escrita terá o formato por exemplo%QX10.1. Porém, se o tipo de dado MODBUS for Holding Register (16 bits), o endereço inicial das variáveis IEC de escrita terá o formato %QW. Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Máscara de Escrita do Register: A função Máscara de Escrita (FC 22), através de uma lógica entre o valor já escrito e as duas words configuradas neste campo, sendo a %QW(0) para a máscara AND e a %QW(2) para a máscara OR; permite ao usuário manipular a word. Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Padrão de Fábrica: O padrão de fábrica não pode ser definido para os campos Área de Diagnósticos do Mapeamento, Variáveis IEC de Leitura, Variável IEC de Escrita e Máscara de Escrita das

Variáveis IEC, pois a criação de uma relação pode ser realizada em qualquer momento no desenvolvimento do aplicativo, fazendo com que o próprio software MasterTool IEC XE aloque um valor, da faixa de variáveis de saída de representação direta (%Q), ainda não utilizado. O padrão de fábrica não pode ser definido para os campos Tamanho de Dados de Leitura e Tamanho de Dados de Escrita, pois eles vão variar de acordo com o tipo de dado MODBUS selecionado.

ATENÇÃO: Diferentemente de outras tarefas de uma aplicação, quando for atingida uma marca de depuração na MainTask, a tarefa de uma instância MODBUS RTU Mestre, e qualquer outra tarefa MODBUS, irá parar de ser executada no momento em que tentar efetuar uma escrita em uma área de memória. Isto ocorre para manter a consistência dos dados das áreas de memória enquanto a MainTask não estiver em execução.

MODBUS RTU Escravo

Este protocolo está disponível para as UCPs da Série Nexto nos seus canais seriais. Ao selecionar esta opção no MasterTool IEC XE, a UCP passa a ser escravo da comunicação MODBUS, permitindo a conexão com dispositivos mestre MODBUS RTU. Este protocolo somente está disponível quando a UCP estiver em modo de execução (Modo Run).

Há dois modos de configuração para este protocolo. Um faz uso de representação direta (%Q), no qual as variáveis são definidas pelo seu endereço. O outro, através de mapeamento simbólico, tem as variáveis definidas pelo seu nome.

Independente do modo de configuração, os passos para inserir uma instância do protocolo e configurar a interface serial são iguais. O procedimento para inserir uma instância de protocolo é encontrado com detalhes no Manual de Manual de Utilização do MasterTool IEC XE – MU299048. As demais etapas de configuração serão descritas a seguir para cada modalidade.

- Adicionar a instância do protocolo MODBUS RTU Escravo ao canal serial COM 1 ou COM 2 (ou ambos, em caso de duas redes de comunicação). Para realizar esse procedimento, consultar o capítulo Programação Inicial – Inserindo uma Instância de Protocolo.
- Configurar a interface serial, escolhendo a velocidade de comunicação, o comportamento dos sinais RTS/CTS, a paridade, os bits de parada do canal, entre outros.
- Consultar o capítulo Configuração - Configuração das Interfaces Seriais.

Configuração do Protocolo MODBUS Escravo por Mapeamento Simbólico

Para configurar este protocolo usando Mapeamento Simbólico, é necessário executar os seguintes passos:

- Configurar os parâmetros gerais do protocolo MODBUS escravo, como: endereço do escravo e tempos de comunicação (disponível no botão de configurações avançadas do Escravo).
- Adicionar e configurar as relações MODBUS, especificando o nome da variável, tipo de dado MODBUS, o endereço inicial do dado e automaticamente são preenchidos o tamanho do dado e a faixa de acordo com o tipo da variável declarada.

Parâmetros Gerais do Protocolo MODBUS Slave – configuração por Mapeamento Simbólico

Os parâmetros gerais, encontrados na tela inicial de configuração do protocolo MODBUS como apresentado na Figura 4-19.

Figura 4-19. Configuração do Escravo

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Endereço do Escravo	Endereço do escravo MODBUS	1	1 a 255

Tabela 4-33. Configurações do Escravo

Os tempos de comunicação do protocolo MODBUS escravo, encontrados no botão "Avançado..." da tela de configuração, estão divididos em: Ciclo da Tarefa, Atraso do Envio e Interframe Mínimo como pode ser visto na Figura 4-20 e na Tabela 4-34.

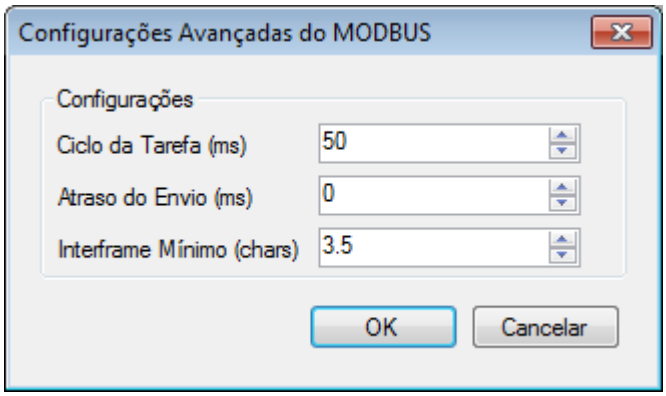

Figura 4-20. Configurações Avançadas MODBUS do Escravo

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Ciclo da Tarefa (ms)	Tempo para execução da instância dentro do ciclo, sem considerar o tempo de execução da mesma	50	20 a 100
Atraso do Envio (ms)	Tempo de atraso para envio da resposta	0	0 a 65535
Interframe Mínimo(chars)	Tempo mínimo de silêncio entre diferentes frames	3.5	3.5 a 100.0

Tabela 4-34. Configurações Avançadas MODBUS Escravo

Notas:

Ciclo da Tarefa: O usuário deverá ter cuidado ao alterar esse parâmetro, pois o mesmo interfere diretamente no tempo de resposta, volume de dados por varredura e, principalmente, no balanceamento dos recursos da UCP entre comunicações e outras tarefas.

Atraso do Envio: A resposta à uma requisição MODBUS pode causar problemas em certos momentos, como, por exemplo, na interface RS-485 ou outra half-duplex. Às vezes existe um atraso entre o tempo da requisição do mestre e o silêncio na linha física (atraso no mestre para colocar RTS em zero e colocar o transmissor RS-485 em alta impedância). Para resolver o problema, o escravo pode esperar o tempo determinado nesse campo antes de enviar a resposta. Caso contrário, os primeiros bytes transmitidos pelo escravo, durante a resposta, podem ser perdidos.

Interframe Mínimo: A norma MODBUS define esse tempo como 3,5 caracteres, porém esse parâmetro é configurável para atender aos dispositivos que não estão de acordo com o padrão.

Os diagnósticos e comandos do protocolo MODBUS Escravo configurado, seja por mapeamento simbólico ou por representação direta, são armazenados em variáveis do tipo T_DIAG_MODBUS_RTU_SLAVE_1 e ainda para o mapeamento por representação direta estão em 4 bytes e 8 words, as quais estão descritas na Tabela 4-35 (n é o valor configurado no campo Endereço Inicial de Diagnósticos em %Q):

Variável de Representação Direta	Variável de diagnósticos do tipo T_DIAG_MODBUS_RTU_SLAVE_1 *.	Tamanho	Descrição

Bits de diagnóstico:				
%QX(n).0	tDiag.*	bRunning	BIT	O escravo está em execução.
%QX(n).1		bNotRunning	BIT	O escravo não está em execução (ver bit: bInterruptedByCommand).
%QX(n).2		bInterruptedByCommand	BIT	O bit bNotRunning foi habilitado, pois o escravo foi interrompido pelo usuário através de bits de comando.
%QX(n).3		bConfigFailure	BIT	Diagnóstico descontinuado.
%QX(n).4		bRXFailure	BIT	Diagnóstico descontinuado.
%QX(n).5		bTXFailure	BIT	Diagnóstico descontinuado.
%QX(n).6		bModuleFailure	BIT	Diagnóstico descontinuado.
%QX(n).7		bDiag_7_reserved	BIT	Reservado.
Códigos de erro:				
%QB(n+1)	eErrorCode		SERIAL_STATUS (BYTE)	0: não existem erros. 1: porta serial inválida. 2 :modo da porta serial inválido 3: baud rate inválido. 4: data bits inválido. 5: paridade inválida. 6: stop bits inválido. 7: parâmetro de sinal de modem inválido. 8: parâmetro de Threshold de RX da UART inválido. 9: parâmetro de time-out inválido. 10: porta serial ocupada. 11: erro de hardware na UART. 12: erro de hardware remoto. 20: tamanho do buffer de transmissão inválido. 21: método de sinal de modem inválido. 22: time-out de CTS = verdadeiro. 23: time-out de CTS = falso. 24: erro de time-out na transmissão. 30: tamanho do buffer de recepção inválido. 31: erro de time-out na recepção. 32: controle de fluxo configurado diferente de manual. 33: controle de fluxo inválido para a porta serial configurada. 34: recepção de dados não permitida no modo normal. 35: recepção de dados não permitida no modo estendido. 36: interrupção DCD não permitida. 37: interrupção CTS não permitida. 38: interrupção DSR não permitida. 39: porta serial não configurada. 50: erro interno na porta serial
Bits de comando, reiniciados automaticamente:				
%QX(n+2).0	tCommand.*	bStop	BIT	Parar o escravo.
%QX(n+2).1		bRestart	BIT	Reiniciar o escravo.
%QX(n+2).2		bResetCounter	BIT	Reiniciar as estatísticas dos diagnósticos (contadores).
%QX(n+2).3		bDiag_19_reserved	BIT	Reservado.
%QX(n+2).4		bDiag_20_reserved	BIT	Reservado.
%QX(n+2).5		bDiag_21_reserved	BIT	Reservado.

%QX(n+2).6		bDiag_22_reserved	BIT	Reservado.
%QX(n+2).7		bDiag_23_reserved	BIT	Reservado.
%QB(n+3)	byDiag_03_reserved		BYTE	Reservado.
Estatísticas de comunicação:				
%QW(n+4)	tStat.*	wRXRequests	WORD	Contador de requisições normais recebidas pelo escravo e respondidas normalmente. No caso de um comando broadcast, esse contador é incrementado, porém não é transmitida a resposta (0 a 65535).
%QW(n+6)		wTXExceptionResponses	WORD	Contador de requisições normais recebidas pelo escravo e respondidas com códigos de exceção. No caso de um comando broadcast, esse contador é incrementado, porém não é transmitida a resposta (0 a 65535). Códigos de exceção: 1: o código da função (FC) é legal, porém não suportado. 2: relação não encontrada nestes dados MODBUS. 3: valor ilegal para o endereço. 128: o mestre/cliente não tem direito de escrita ou leitura. 129: a relação MODBUS está desabilitada.
%QW(n+8)		wRXFrames	WORD	Contador de frames recebidos pelo escravo. Considera-se frame, algo que é processado e seguido de um período mínimo de silêncio interframes (Interframe Mínimo), ou seja, uma mensagem ilegal também é computada (0 a 65535).
%QW(n+10)		wRXIllegalRequests	WORD	Contador de requisições ilegais. Estes são frames que iniciam com o endereço 0 (broadcast) ou com o endereço MODBUS do escravo, mas não são requisições legais – sintaxe inválida, frames menores, CRC inválido – (0 a 65535).
%QW(n+12)		wRXOverrunErrors	WORD	Contador de frames com erros de overrun durante a recepção – UART FIFO ou fila RX – (0 a 65535).
%QW(n+14)		wRXIncompleteFrames	WORD	Contador de frames com erro de construção, paridade ou falha durante a recepção (0 a 65535).
%QW(n+16)		wCTSTimeOutErrors	WORD	Contador de erro de time-out no CTS, utilizando o handshake RTS/CTS, durante a transmissão (0 a 65535).
%QW(n+18)		wDiag_18_reserved	WORD	Reservado.

Tabela 4-35. Diagnósticos MODBUS RTU Slave

Nota:

Contadores: Todos os contadores dos diagnósticos do MODBUS RTU Escravo retornam à zero quando o valor limite 65535 é ultrapassado.

Configuração dos Mapeamentos – Configuração por Mapeamento Simbólico

A configuração dos mapeamentos MODBUS, visualizadas na Figura 4-21, segue os parâmetros descritos na Tabela 4-36:

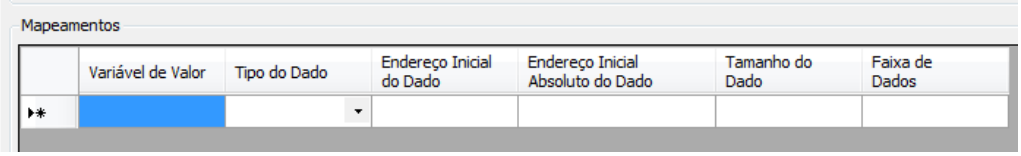

Figura 4-21. Tela de Mapeamentos de dados MODBUS

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Variável de Valor	Nome da variável simbólica	-	Nome de uma variável declarada em um programa ou GVL
Tipo do Dado	Tipo de dado MODBUS	-	Coil (1 bit) Input Status (1 bit) Holding Register (16 bits) Input Register (16 bits)
Endereço Inicial do Dado	Endereço inicial dos dados MODBUS	-	1 a 65536
Tamanho do Dado	Tamanho do dado MODBUS	-	1 a 65536
Faixa de Dados	Faixa de endereços do dado configurado	-	-

Tabela 4-36. Configuração dos Mapeamentos MODBUS

Notas:

Variável de Valor: Campo é utilizado para especificar uma variável simbólica na relação MODBUS.

Tipo do Dado: Campo é utilizado para especificar o tipo de dado utilizado na relação MODBUS.

Tipo de Dado	Tamanho [bits]	Descrição
Coil	1	Saída digital que pode ser lida ou escrita.
Input Status	1	Entrada digital que pode ser apenas lida.
Holding Register	16	Saída analógica que pode ser lida ou escrita.
Input Register	16	Entrada analógica que pode ser apenas lida.

Tabela 4-37. Tipos de Dados suportados no MODBUS RTU Escravo

Endereço Inicial do Dado: Endereço inicial do dado de uma relação MODBUS.

Tamanho do Dado: O valor de tamanho especifica a quantidade máxima de dados que uma relação MODBUS poderá acessar, a partir do endereço inicial. Sendo assim, para ler uma faixa de endereços contínua, é necessário que todos os endereços estejam declarados em uma única relação. Este campo varia de acordo com o tipo de dado MODBUS configurado.

Faixa de Dados: Este campo mostra ao usuário a faixa de endereços de memória utilizada pela relação MODBUS.

ATENÇÃO:
Diferentemente de outras tarefas de uma aplicação, quando for atingida uma marca de depuração na MainTask, a tarefa de uma instância MODBUS RTU Escravo, e qualquer outra tarefa MODBUS, irá parar de ser executada no momento em que tentar efetuar uma escrita em uma área de memória. Isto ocorre para manter a consistência dos dados das áreas de memória enquanto a MainTask não estiver em execução.

Configuração do Protocolo MODBUS Escravo por Representação Direta (%Q)

Para configurar este protocolo usando Representação Direta (%Q), é necessário executar os seguintes passos:

- Configurar os parâmetros gerais do protocolo MODBUS escravo, como: tempos de comunicação, endereço e variáveis de representação direta (%Q) para receber os diagnósticos e controlar as relações.
- Adicionar e configurar relações MODBUS, especificando o tipo de dado MODBUS, variáveis de representação direta (%Q) para receber/escrever os dados e quantidade de dados a comunicar.

As descrições de cada configuração estão relacionadas a seguir, neste capítulo.

Parâmetros Gerais do Protocolo MODBUS Escravo – Configuração por Representação Direta (%Q)

Os parâmetros gerais, encontrados na tela inicial de configuração do protocolo MODBUS (Figura 4-22), são definidos como:

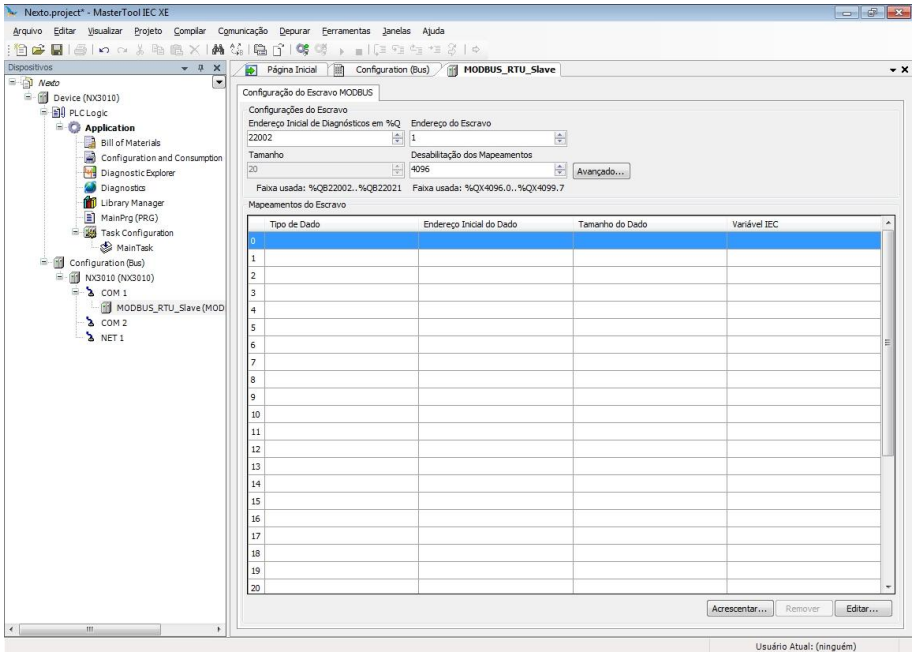

Figura 4-22. Tela de Configuração MODBUS RTU Escravo

- Endereço, Variáveis de representação direta (%Q) para controlar as relações e os diagnósticos:

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Endereço Inicial de Diagnósticos em %Q	Endereço inicial das variáveis de diagnóstico	-	0 a 2147483628
Tamanho	Tamanho da área de diagnósticos	-	Desabilitado para edição
Endereço do Escravo	Endereço do escravo MODBUS	1	1 a 255
Desabilitação dos Mapeamentos	Endereço inicial utilizado para desabilitar as relações MODBUS	-	0 a 2147483644

Tabela 4-38. Configurações

Notas:

Endereço Inicial de Diagnósticos em %Q: Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Endereço do Escravo: É importante salientar que o escravo aceita requisições broadcast, quando o mestre envia um comando com o endereço configurado como zero. Além disso, de acordo com a

Norma MODBUS, a faixa de endereços válidos para escravos é de 1 a 247, sendo os endereços 248 a 255 reservados.

Desabilitação dos Mapeamentos: Composto por 32 bits, utilizados para desabilitar, individualmente, as 32 relações MODBUS configuradas no espaço Mapeamentos do Escravo. A relação é desabilitada quando o bit, correspondente a relação, for igual a 1, caso contrário, o mapeamento está habilitado. Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Padrão de Fábrica: O padrão de fábrica não pode ser definido para os campos Endereço Inicial de diagnósticos em %Q e Desabilitação dos Mapeamentos, pois a criação de uma instância do protocolo pode ser realizada em qualquer momento no desenvolvimento do aplicativo, fazendo com que o próprio software MasterTool IEC XE aloque um valor, da faixa de variáveis de saída de representação direta (%Q), ainda não utilizado.

Os diagnósticos e comandos do protocolo MODBUS estão descritoas na Tabela 4-35

Os tempos de comunicação do protocolo MODBUS Escravo, encontrados no botão “Avançado...” da tela de configuração, estão descritos naTabela 4-34.

Configuração dos Mapeamentos – Configuração por Representação Direta (%Q)

A configuração das relações MODBUS, visualizada na Figura 4-23 e Figura 4-24, segue os parâmetros descritos na Tabela 4-39:

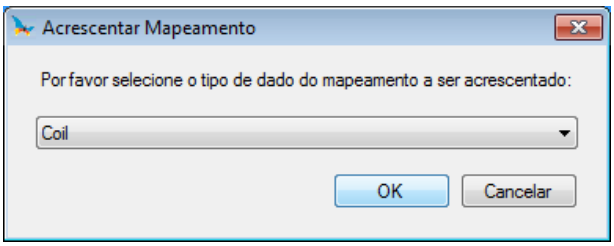

Figura 4-23. Adicionando Relações MODBUS

Figura 4-24. Configurando a Relação MODBUS

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Tipo de Dado	Tipo de dado MODBUS	Coil	Coil (1 bit) Holding Register (16 bits) Input Status (1 bit) Input Register (16 bits)
Endereço Inicial do Dado	Endereço inicial dos dados MODBUS	1	1 a 65536
Tamanho do Dado	Número de dados MODBUS	-	1 a 65536
Variável IEC	Endereço inicial das variáveis (%Q)	-	0 a 2147483647
Somente Leitura	Somente permite a leitura	Desabilitada	Habilitada ou Desabilitada

Tabela 4-39. Mapeamentos do Escravo

Notas:

Possibilidades: Os valores descritos na coluna Possibilidades podem variar de acordo com o tipo de dado MODBUS configurado.

Tamanho do Dado: O valor de Tamanho do Dado define a quantidade máxima de dados que uma relação MODBUS poderá acessar, a partir do endereço inicial. Sendo assim, para ler uma faixa de endereços contínua, é necessário que todos os endereços estejam declarados em uma única relação. Este campo varia de acordo com o tipo de dado MODBUS configurado, ou seja, quando selecionado tipo Coil ou Input Status, o campo data size deve ser um número múltiplo de oito. Também deve-se dar atenção para que o valor máximo não seja superior ao tamanho da memória de saídas endereçáveis e não sejam atribuídos os mesmos valores já utilizados durante a aplicação.

Variável IEC: Caso o tipo de dado MODBUS seja Coil ou Input Status (bit), o endereço inicial das variáveis IEC terá o formato por exemplo %QX10.1. Porém, se o tipo de dado MODBUS for Holding Register ou Input Register (16 bits), o endereço inicial das variáveis IEC terá o formato %QW. Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Somente Leitura: Quando habilitada, somente permite que o mestre da comunicação leia os dados das variáveis, não permitindo a escrita. Opção válida somente para as funções de escrita.

Padrão de Fábrica: O padrão de fábrica não pode ser definido para o campo Variável IEC, pois a criação de uma instância do protocolo pode ser realizada em qualquer momento no desenvolvimento do aplicativo, fazendo com que o próprio software MasterTool IEC XE aloque um valor, da faixa de variáveis de saída de representação direta (%Q), ainda não utilizado. O padrão de fábrica não pode ser definido para o campo Tamanho do Dado, pois ele vai variar de acordo com o tipo de dado MODBUS selecionado.

Nas relações definidas anteriormente, o tamanho máximo de dados MODBUS pode ser 65535 (máximo valor configurado no campo Tamanho do Dado). Porém, a pergunta que chega no MODBUS RTU Escravo deverá endereçar um subconjunto desse mapeamento e esse grupo deve ter, no máximo, o tamanho de dados que depende do código da função, os quais estão definidos abaixo:

- Leitura de Coils (FC 1): 2000
- Leitura de Input Status (FC 2): 2000
- Leitura de Holding Registers (FC 3): 125
- Leitura de Input Registers (FC 4): 125
- Escrita de Single Coil (FC 5): 1
- Escrita de Single Holding Register (FC 6): 1
- Forçamento de Multiple Coils (FC 15): 1968
- Escrita de Holding Registers (FC 16): 123

- Máscara de Escrita do Register (FC 22): 1
- Leitura/Escrita de Holding Registers (FC 23):
  - Leitura: 121
  - Escrita: 121

ATENÇÃO: Diferentemente de outras tarefas de uma aplicação, quando for atingida uma marca de depuração na MainTask, a tarefa de uma instância MODBUS RTU Escravo, e qualquer outra tarefa MODBUS, irá parar de ser executada no momento em que tentar efetuar uma escrita em uma área de memória. Isto ocorre para manter a consistência dos dados das áreas de memória enquanto a MainTask não estiver em execução.

MODBUS Ethernet

A rede de comunicação multimestre permite que as UCPs Nexto leiam ou escrevam variáveis MODBUS em outros controladores ou IHMs compatíveis com os protocolos MODBUS TCP ou MODBUS RTU via TCP. A UCP Nexto pode, simultaneamente, ser cliente e servidor em uma mesma rede de comunicação, ou até mesmo ter mais instâncias associadas à interface Ethernet, indiferente se as mesmas são MODBUS TCP ou MODBUS RTU via TCP, conforme descreve a Tabela 4-17.

A Figura 4-25 representa algumas das possibilidades de comunicação utilizando-se o protocolo MODBUS TCP simultaneamente com o protocolo MODBUS RTU via TCP.

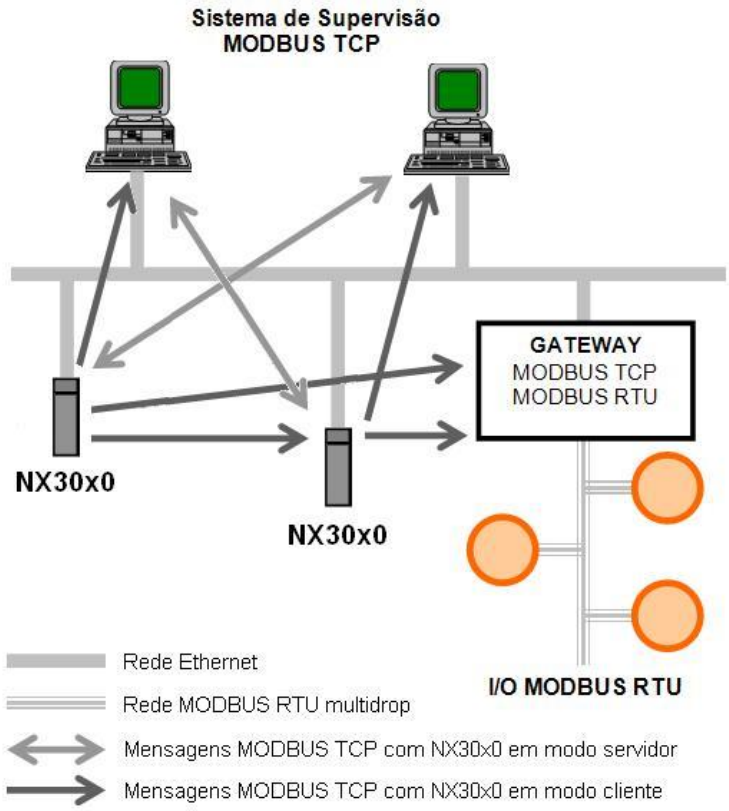

Figura 4-25. Rede de Comunicação MODBUS TCP

A associação de variáveis MODBUS com variáveis simbólicas da UCP é realizada pelo usuário através da definição de relações via configurador MasterTool IEC XE. Podem ser definidas até 32 relações para o modo servidor e até 128 relações para o modo cliente. Uma relação, em modo servidor, pode definir uma grande área de dados MODBUS e torná-la disponível para vários clientes.

As relações em modo cliente, por outro lado, devem respeitar o tamanho máximo de dados de uma função MODBUS: 125 registradores (input registers ou holding registers) ou 2000 bits (coils ou input status). Essas informações são detalhadas na descrição de cada protocolo.

Todas as relações, em modo cliente ou servidor, podem ser desabilitadas através de variáveis de representação direta (%Q) identificadas como Desabilitação dos Mapeamentos pelo MasterTool IEC XE. A desabilitação pode ocorrer através de bits gerais, os quais afetam todas as relações de um modo de operação, ou através de bits específicos, afetando relações específicas.

Para as relações em modo servidor, podem ser definidos conjuntos de endereços IPs com permissão de escrita e leitura, chamados de filtros. Isto é feito através da definição de um endereço de rede IP e de uma máscara de subrede, resultando em um grupo de IPs clientes que podem escrever e ler nas variáveis da relação. Funções de leitura/escrita são filtradas da mesma forma que as funções exclusivas de leitura ou escrita. Essas informações são detalhadas na descrição do protocolo MODBUS Ethernet Servidor.

Quando o protocolo MODBUS TCP é utilizado no modo cliente, pode-se usufruir da característica de múltiplas requisições, utilizando a mesma conexão TCP para acelerar a comunicação com os servidores. Quando esta característica não for desejada ou não for suportada pelo servidor, ela pode ser desabilitada (ação em nível de relação). É importante destacar que o número máximo de conexões TCP entre cliente e servidor é 63, sendo que se alguns parâmetros forem alterados, comunicações inativas podem ser fechadas, possibilitando a abertura de novas conexões.

A Tabela 4-40 e a Tabela 4-41 trazem, respectivamente, a lista completa dos tipos de dados e funções MODBUS suportadas pelas UCPs Nexto.

Tipo de Dado	Tamanho [bits]	Descrição
Coil	1	Saída digital que pode ser lida ou escrita.
Input Status	1	Entrada digital que pode ser apenas lida.
Holding Register	16	Saída analógica que pode ser lida ou escrita.
Input Register	16	Entrada analógica que pode ser apenas lida.

Tabela 4-40. Tipos de Dados MODBUS Suportados pelas UCPs Nexto

Tipo de Função	Código		Descrição
	DEC	HEX	
Acesso a Variáveis	1	0x01	Leitura de coils (FC 1)
	2	0x02	Leitura de input status (FC 2)
	3	0x03	Leitura de holding registers (FC 3)
	4	0x04	Leitura de input registers (FC 4)
	5	0x05	Escrita de um coil (FC 5)
	6	0x06	Escrita de um holding register (FC 6)
	15	0x0F	Escrita de múltiplos coils (FC 15)
	16	0x10	Escrita de múltiplos holding registers (FC 16)
	22	0x16	Escrita mascarada de um holding register (FC 22)
	23	0x17	Leitura/escrita de múltiplos holding registers (FC 23)

Tabela 4-41. Funções MODBUS Suportadas pelas UCPs Nexto

Independente do modo de configuração, os passos para inserir uma instância do protocolo e configurar a interface ethernet são iguais. As demais etapas de configuração serão descritas a seguir para cada modalidade.

- Adicionar uma ou mais instâncias do protocolo MODBUS Ethernet Cliente ou Servidor ao canal Ethernet NET 1 ou NET 2 (ou ambos, em caso de mais de uma rede de comunicação). Para realizar esse procedimento, consultar o capítulo Programação Inicial – Inserindo uma Instância de Protocolo.

- Configurar a interface Ethernet. Para realizar esse procedimento, consultar o capítulo Configuração – Configuração das Interfaces Ethernet.

MODBUS Ethernet Cliente

Este protocolo está disponível para as UCPs da Série Nexto nos seus canais Ethernet. Ao selecionar esta opção no MasterTool IEC XE, a UCP passa a ser cliente da comunicação MODBUS, possibilitando o acesso a outros dispositivos com o mesmo protocolo, quando esta estiver em modo de execução (Modo Run).

Há dois modos de configuração para este protocolo. Um deles faz uso de Representação Direta (%Q), no qual as variáveis são definidas pelo seu endereço. O outro, chamado Mapeamento Simbólico, tem as variáveis definidas pelo seu nome.

O procedimento para inserir uma instância de protocolo é encontrado com detalhes no Manual de Manual de Utilização do MasterTool IEC XE – MU299048 ou no capítulo Inserindo uma Instância de Protocolo.

Configuração do Protocolo MODBUS Ethernet Cliente por Mapeamento Simbólico

Para configurar este protocolo usando Mapeamento Simbólico, é necessário executar os seguintes passos:

- Configurar os parâmetros gerais do protocolo MODBUS Cliente, com o protocolo TCP ou RTU via TCP.
- Adicionar e configurar dispositivos, definindo endereço IP, porta, endereço do escravo e time-out de comunicação (disponível no botão de configurações avançadas do Dispositivo).
- Adicionar e configurar os mapeamentos MODBUS, especificando o nome da variável, tipo de dados, endereço inicial do dado, tamanho do dado e variável que receberá os dados de qualidade.
- Adicionar e configurar as requisições MODBUS, especificando a função desejada, o tempo de varredura da requisição, o endereço inicial (leitura/escrita), o tamanho dos dados (Leitura/Escrita), a variável que receberá os dados de qualidade, e a variável responsável por desabilitar a requisição.

Parâmetros Gerais do Protocolo MODBUS Cliente – Configuração por Mapeamento Simbólico

Os parâmetros gerais, encontrados na tela inicial de configuração do protocolo MODBUS (Figura 4-26), são definidos como:

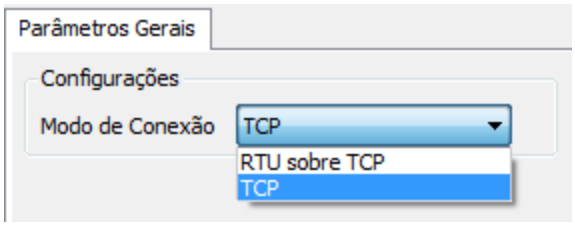

Figura 4-26. Tela de Configuração Parâmetros Gerais MODBUS Cliente

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Modo de Conexão	Seleção do protocolo	TCP	RTU via TCP TCP

Tabela 4-42. Configurações Gerais MODBUS Cliente

Os diagnósticos e comandos do protocolo MODBUS Cliente configurado, seja por mapeamento simbólico ou por representação direta, são armazenados em variáveis do tipo T_DIAG_MODBUS_ETH_CLIENT_1 e ainda para o mapeamento por representação direta estão em 4 bytes e 8 word, as quais estão descritas na Tabela 4-43(n é o valor configurado no campo Endereço Inicial de Diagnósticos em %Q):

Variável de Representação Direta	Variável de diagnóstico do tipo T_DIAG_MODBUS_ETH_CLIENT_1.*		Tamanho	Descrição
Bits de diagnóstico:				
%QX(n).0	tDiag.*	bRunning	BIT	O cliente está em execução.
%QX(n).1		bNotRunning	BIT	O cliente não está em execução (ver bit bInterruptedByCommand).
%QX(n).2		bInterruptedByCommand	BIT	O bit bNotRunning foi habilitado pois o cliente foi interrompido pelo usuário através de bits de comando.
%QX(n).3		bConfigFailure	BIT	Diagnóstico descontinuado.
%QX(n).4		bRXFailure	BIT	Diagnóstico descontinuado.
%QX(n).5		bTXFailure	BIT	Diagnóstico descontinuado.
%QX(n).6		bModuleFailure	BIT	Indica se há falha no módulo ou o módulo não está presente.
%QX(n).7		bAllDevicesCommFailure	BIT	Indica que todos os dispositivos configurados no cliente estão com falha.
%QB(n+1)	byDiag_1_reserved		BYTE	Reservado.
Bits de comando, reiniciados automaticamente:				
%QX(n+2).0	tCommand.*	bStop	BIT	Parar o cliente.
%QX(n+2).1		bRestart	BIT	Reiniciar o cliente.
%QX(n+2).2		bResetCounter	BIT	Reiniciar as estatísticas dos diagnósticos (contadores).
%QX(n+2).3		bDiag_19_reserved	BIT	Reservado.
%QX(n+2).4		bDiag_20_reserved	BIT	Reservado.
%QX(n+2).5		bDiag_21_reserved	BIT	Reservado.
%QX(n+2).6		bDiag_22_reserved	BIT	Reservado.
%QX(n+2).7		bDiag_23_reserved	BIT	Reservado.
%QB(n+3)	byDiag_03_reserved		BYTE	Reservado.
Estatísticas de comunicação:				
%QW(n+4)	tStat.*	wTXRequests	WORD	Contador de requisições transmitidas pelo cliente (0 a 65535).
%QW(n+6)		wRXNormalResponses	WORD	Contador de respostas normais recebidas pelo cliente (0 a 65535).
%QW(n+8)		wRXExceptionResponses	WORD	Contador de respostas com códigos de exceção (0 a 65535).
%QW(n+10)		wRXIllegalResponses	WORD	Contador de respostas ilegais recebidas pelo cliente – sintaxe inválida, CRC inválido ou número insuficiente de bytes recebidos (0 a 65535).
%QW(n+12)		wDiag_12_reserved	WORD	Reservado.
%QW(n+14)		wDiag_14_reserved	WORD	Reservado.
%QW(n+16)		wDiag_16_reserved	WORD	Reservado.
%QW(n+18)		wDiag_18_reserved	WORD	Reservado.

Tabela 4-43. Diagnósticos MODBUS Cliente

Nota:

Contadores: Todos os contadores dos diagnósticos do MODBUS TCP Cliente retornam à zero quando o valor limite 65535 é ultrapassado.

Configuração dos Dispositivos – Configuração por Mapeamento Simbólico

A configuração dos dispositivos escravos, visualizados na Tabela 4-44, seguem os seguintes parâmetros:

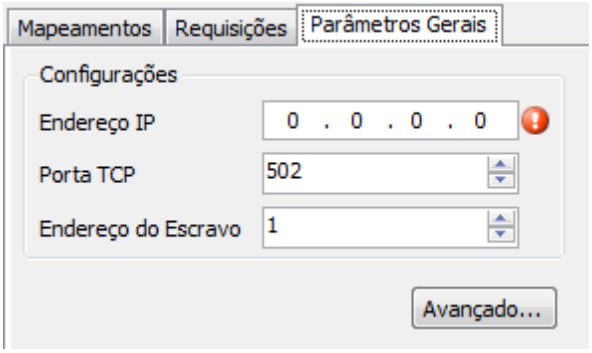

Figura 4-27. Tela de Configurações dos Parâmetros Gerais do Dispositivo

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Endereço IP	Endereço IP do servidor	0.0.0.0	1.0.0.1 a 223.255.255.255
Porta TCP	Porta TCP	502	2 a 65534
Endereço do Escravo	Endereço do Escravo MODBUS	1	0 a 255

Tabela 4-44. Configurações Gerais MODBUS Cliente

Notas:

Endereço IP: Endereço IP do dispositivo servidor MODBUS.

Porta TCP: Caso sejam adicionadas várias instâncias do protocolo em uma única interface Ethernet, diferentes portas TCP devem ser selecionadas para cada instância. Algumas portas TCP, entre as possibilidades mencionadas acima, são reservadas e, portanto, não podem ser utilizadas. São elas: 80, 8080, 1217, 1740, 1741, 1742,1743 e 11740.

Endereço do Escravo: De acordo com a norma MODBUS, a faixa de endereços válidos para escravos é de 0 a 247, sendo os endereços 248 a 255 reservados. Quando o mestre envia um comando de escrita com o endereço configurado como zero, ele está realizando requisições broadcast na rede.

Os parâmetros nas configurações avançadas do dispositivo MODBUS Cliente, encontrados no botão "Avançado..." na aba de Parâmetros Gerais, estão divididos em: número máximo de requisições simultâneas, Time-out de Comunicação, Modo de time-out de conexão e tempo de inatividade.

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Número Máximo de Requisições Simultâneas	Número de requisições simultâneas que o cliente pode fazer ao servidor	1	1 a 8
Time-out de Comunicação	Time-out do nível da aplicação em ms	3000	10 a 65535
Modo	Define quando a conexão com o servidor é finalizada pelo cliente	Conexão é fechada após um tempo de inatividade de (s): 10 a 3600.	Conexão é fechada depois de um time-out. Conexão é fechada ao final de cada comunicação. Conexão é fechada após um tempo de inatividade de (s): 10 a 3600.
Tempo de Inatividade	Tempo de inatividade	10	3600

Tabela 4-45. Configurações Avançadas MODBUS Cliente

Notas:

Número Máximo de Requisições Simultâneas: É utilizado em servidores com um alto ciclo de varredura. Esse parâmetro é fixado em 1 (não editável), quando o protocolo configurado é MODBUS RTU via TCP.

Time-out de Comunicação: O time-out da comunicação é o tempo que o cliente aguardará por uma resposta do servidor à requisição. Para um dispositivo MODBUS Cliente, duas variáveis do sistema devem ser consideradas: o tempo que o servidor leva para processar a requisição e o atraso de envio da resposta caso seja configurado no servidor. É recomendado que o time-out seja igual ou maior que duas vezes a soma destes parâmetros. Para mais informações, ver capítulo Configuração - Configuração de Protocolos - Desempenho de Comunicação.

Modo: Define quando a conexão com o servidor é finalizada pelo cliente. Seguem as opções:

- *Conexão é fechada depois de um time-out* ou *Conexão nunca é fechada em situações normais*: Estas opções apresentam o mesmo comportamento do Cliente fechar a conexão devido ao fato do Servidor não ter respondido a uma requisição antes do *Time-out de Comunicação* ter se esgotado.
- *Conexão é fechada ao final de cada comunicação*: A conexão é fechada pelo Cliente após concluir cada requisição.
- *Conexão é fechada após um tempo de inatividade:* A conexão será fechada pelo Cliente caso ele fique por um tempo igual ao *Tempo de Inatividade* sem realizar requisição para o Servidor.

Tempo de Inatividade: Tempo de inatividade da conexão.

Configuração dos Mapeamentos – configuração por Mapeamento Simbólico

A configuração dos mapeamentos MODBUS Cliente, visualizadas na Figura 4-28, segue os parâmetros descritos na Tabela 4-46:

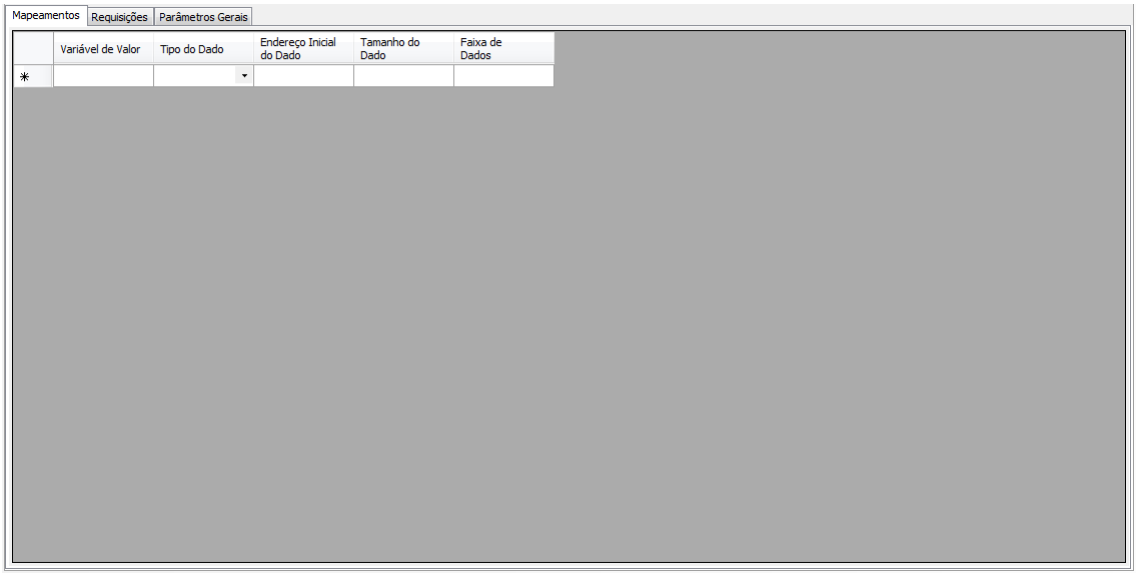

Figura 4-28. Tela de Mapeamentos de dados MODBUS Cliente

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Variável de Valor	Nome da variável simbólica	-	Nome de uma variável declarada em um programa ou GVL
Tipo do Dado	Tipo de dado MODBUS	-	Coil Escrita(1 bit) Coil Leitura (1 bit) Holding Register Escrita (16 bits) Holding Register Leitura (16 bits) Holding Register – Máscara AND (16 bits) Holding Register – Máscara OR (16 bits) Input Register (16 bits)

			Input Status (1 bit)
Endereço Inicial do Dado	Endereço inicial dos dados MODBUS	-	1 a 65536
Tamanho do Dado	Tamanho do dado MODBUS	-	1 a 65536
Faixa de Dados	Faixa de endereços do dado configurado	-	-

Tabela 4-46. Configuração dos Mapeamentos MODBUS Cliente

Notas:

Variável de Valor: Esse campo é utilizado para especificar uma variável simbólica na relação MODBUS.

Tipo do Dado: Esse campo é utilizado para especificar o tipo de dado utilizado na relação MODBUS.

Tipo de Dado	Tamanho [bits]	Descrição
Coil Escrita	1	Saída digital de escrita.
Coil Leitura	1	Saída digital de leitura.
Holding Register de Escrita	16	Saída analógica de escrita.
Holding Register de Leitura	16	Saída analógica de leitura.
Holding Register - Máscara AND	16	Máscara AND. Utilizada na escrita de saída analógica.
Holding Register -Máscara OR	16	Máscara OR. Utilizada na escrita de saída analógica.
Input Register	16	Entrada analógica que pode ser apenas lida.
Input Status	1	Entrada digital que pode ser apenas lida.

Tabela 4-47. Tipos de Dados suportados no MODBUS Cliente

Endereço Inicial do Dado: Endereço inicial do dado de um mapeamento MODBUS.

Tamanho do Dado: O valor de Tamanho especifica a quantidade máxima de dados que uma relação MODBUS poderá acessar, a partir do endereço inicial. Sendo assim, para ler uma faixa de endereços contínua, é necessário que todos os endereços estejam declarados em uma única relação. Este campo varia de acordo com o tipo de dado MODBUS configurado.

Faixa de Dados: Este campo mostra ao usuário a faixa de endereços de memória utilizada pela relação MODBUS.

Configuração das Requisições – configuração por Mapeamento Simbólico

A configuração das requisições MODBUS, visualizadas na Figura 4-29, segue os parâmetros descritos na Tabela 4-48:

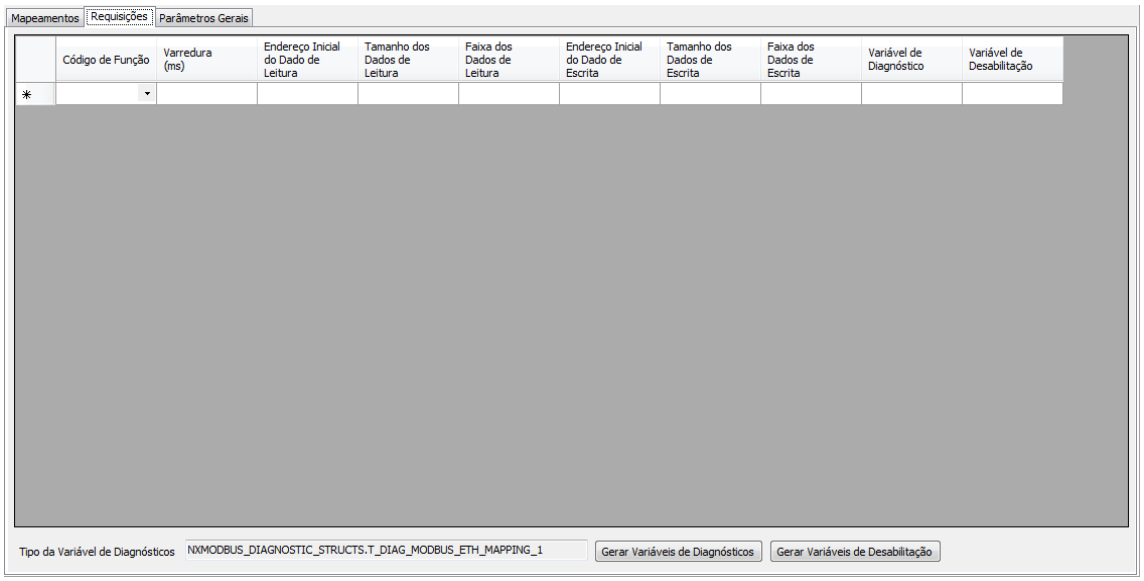

Figura 4-29. Tela de Requisições de dados MODBUS

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Código de Função	Tipo de função MODBUS	-	01 – Leitura de Coils 02 – Leitura de Input Status 03 – Leitura de Holding Registers 04 – Leitura de Input Registers 05 – Escrita de um Coil 06 – Escrita de um Register 15 – Escrita de múltiplos Coils 16 – Escrita de Múltiplos Registers 22 –Escrita Mascarada de Register 23 – Leitura/Escrita de Múltiplos Register
Varredura (ms)	Período de comunicação (ms)	100	0 a 3600000
Endereço Inicial do Dado de Leitura	Endereço inicial dos dados de leitura MODBUS	-	1 a 65536
Tamanho dos Dados de Leitura	Tamanho dos dados de leitura MODBUS	-	Depende da função utilizada
Faixa dos Dados de Leitura	Faixa de endereço dos dados de leitura MODBUS	-	0 a 2147483646
Endereço Inicial do Dado de Escrita	Endereço inicial dos dados de escrita MODBUS	-	1 a 65536
Tamanho dos Dados de Escrita	Tamanho dos dados de escrita MODBUS	-	Depende da função utilizada
Faixa dos dados de Escrita	Faixa de endereço dos dados de escrita MODBUS	-	0 a 2147483647
Variável de Diagnósticos	Nome da variável de diagnóstico	-	Nome de uma variável declarada em um programa ou GVL
Variável de Desabilitação	Variável utilizada para desabilitar a relação MODBUS	-	Campo destinado a variável simbólica utilizada para desabilitar, individualmente, as requisições MODBUS configuradas. Esta variável deve ser do tipo BOOL. A variável pode ser simples ou elemento de array e pode estar em estruturas.

Tabela 4-48. Configuração das Relações MODBUS Cliente

Notas:

Configuração das relações MODBUS: O número de configurações, padrão de fábrica e os valores da coluna possibilidades, podem variar de acordo com o tipo de dado e função MODBUS (FC).

Código de Funções: As funções MODBUS (FC) disponíveis são os seguintes:

Tipo de Função	Código		Descrição
	DEC	HEX	
Acesso a Variáveis	1	0x01	Leitura de coils (FC 01)
	2	0x02	Leitura de input status (FC 02)
	3	0x03	Leitura de holding registers (FC 03)
	4	0x04	Leitura de input registers (FC 04)
	5	0x05	Escrita de um coil (FC 05)
	6	0x06	Escrita de um holding register (FC 06)
	15	0x0F	Escrita de múltiplos coils (FC 15)
	16	0x10	Escrita de holding registers (FC 16)
	22	0x16	Máscara de escrita do register (FC 22)
	23	0x17	Leitura/escrita holding registers (FC 23)

Tabela 4-49. Funções MODBUS Cliente

Varredura: Este parâmetro indica com que frequência à comunicação definida por esta requisição deve ser executada. Ao ser finalizada uma comunicação será aguardado um tempo igual ao configurado no campo varredura e, após, será executada uma nova comunicação.

Endereço Inicial do Dado de Leitura: Campo destinado ao endereço inicial dos dados de leitura MODBUS.

Tamanho dos Dados de Leitura: O valor mínimo para o tamanho dos dados de leitura é 1 e o valor máximo depende da função MODBUS (FC) utilizada, conforme abaixo:

- Leitura de Coils (FC 1): 2000
- Leitura de Input Status (FC 2): 2000
- Leitura de Holding Registers (FC 3): 125
- Leitura de Input Registers (FC 4): 125
- Leitura/Escrita de Holding Registers (FC 23): 121

Faixa dos Dados de Leitura: Este campo mostra a faixa de dados de leitura MODBUS configurada para cada requisição. O endereço inicial de leitura, somado ao tamanho do dado de leitura resultará na faixa de dados de leitura de cada uma das requisições.

Endereço Inicial do Dado de Escrita: Campo destinado ao endereço inicial dos dados de escrita MODBUS.

Tamanho dos Dados de Escrita: O valor mínimo para o tamanho dos dados de escrita é 1 e o valor máximo depende da função MODBUS (FC) utilizada, conforme abaixo:

- Escrita de Um Coil (FC 5): 1
- Escrita de Um Holding Registers (FC 6): 1
- Escrita de Múltiplos Coils (FC 15): 1968
- Escrita de Holding Registers (FC 16): 123
- Máscara de Escrita do Register (FC 22):
- Leitura/Escrita de Holding Registers (FC 23): 121

Faixa dos Dados de Escrita: Este campo mostra a faixa de dados de escrita MODBUS configurada para cada requisição. O endereço inicial de escrita, somado ao tamanho do dado de escrita resultará na faixa de dados de escrita de cada uma das requisições.

Variável de Diagnósticos: Os diagnósticos da requisição MODBUS configurada, seja por mapeamento simbólico ou por representação direta, são armazenados em variáveis do tipo T_DIAG_MODBUS_ETH_CLIENT_1 e ainda para o mapeamento por representação direta estão em 4 bytes e 2 words,a qual está descrita na Tabela 4-50 (n é o valor configurado no campo Endereço Inicial de Diagnósticos em %Q).

Variável de Representação Direta	Variável de diagnóstico do tipo T_DIAG_MODBUS_ETH_MAPPING_1.*		Tamanho	Descrição
Bits de estado da comunicação:				
%QX(n).0	byStatus.*	bCommIdle	BIT	Comunicação inativa (aguardando ser constada).
%QX(n).1		bCommExecuting	BIT	Comunicação ativa.
%QX(n).2		bCommPostponed	BIT	Comunicação adiada, pois o número máximo de requisições simultâneas foi atingido. As comunicações adiadas serão executadas na mesma sequência em que elas foram requisitadas, evitando a indeterminação. O tempo gasto nesse estado não é contabilizado para efeitos de time-out. Os bits bCommIdle e bCommExecuting são falsos quando o bit bCommPostponed é verdadeiro.
%QX(n).3		bCommDisabled	BIT	Comunicação desabilitada. O bit bCommIdle é reiniciado nessa condição.
%QX(n).4		bCommOk	BIT	Comunicação finalizada anteriormente foi realizada com sucesso.
%QX(n).5		bCommError	BIT	Comunicação finalizada anteriormente teve um erro. Verificar código de erro.
%QX(n).6		bCommAborted	BIT	Comunicação finalizada anteriormente foi interrompida devido à falha de conexão.
%QX(n).7		bDiag_7_reserved	BIT	Reservado.
Último código de erro (habilitado quando o bCommError = verdadeiro):				
%QB(n+1)	eLastErrorCode		MASTER_ERROR_CODE (BYTE)	Informa a possível causa do último erro ocorrido na relação MODBUS. Consulte a Tabela 4-51 para detalhes das possibilidades.
Último código de exceção recebido pelo cliente:				
%QB(n+2)	eLastExceptionCode		MODBUS_EXCEPTION (BYTE)	NO_EXCEPTION (0) FUNCTION_NOT_SUPPORTED (1) MAPPING_NOT_FOUND (2) ILLEGAL_VALUE (3) ACCESS_DENIED (128)* MAPPING_DISABLED (129)* IGNORE_FRAME (255)*
Estatísticas de comunicação:				
%QB(n+3)	byDiag_3_reserved		BYTE	Reservado.
%QW(n+4)	wCommCounter		WORD	Contador de comunicações finalizadas, com ou sem erros. O usuário pode testar quando a comunicação foi finalizada testando a variação desse contador. Quando o valor 65535 é atingido, o contador retorna à zero.

%QW(n+6)	wCommErrorCounter	WORD	Contador de comunicações finalizadas com erros. Quando o valor 65535 é atingido, o contador retorna a zero.

Tabela 4-50. Diagnósticos das relações MODBUS Cliente

Códigos de exceção: Os códigos de exceção apresentados neste campo são os valores retornados pelo servidor. As definições dos códigos de exceção 128, 129 e 255 são válidas apenas na utilização de escravos Altus. Para escravos de outros fabricantes esses códigos de exceção podem ter significados diferentes.

Desabilitação: Campo destinado a variável utilizada para desabilitar, individualmente, as requisições MODBUS configuradas no espaço Requisições. A requisição é desabilitada quando a variável, correspondente a requisição, for igual a 1, caso contrário, a requisição está habilitada.

Último código de erro: Os códigos das possíveis situações que ocasionam erro na comunicação MODBUS podem ser consultados abaixo:

Código	Enumerável	Descrição
1	ERR_EXCEPTION	Resposta reportada em um código de exceção (ver eLastExceptionCode = Código de Exceção).
2	ERR_CRC	Resposta com CRC inválido.
3	ERR_ADDRESS	Endereço MODBUS não encontrado. O endereço que respondeu à requisição foi diferente do esperado.
4	ERR_FUNCTION	Código inválido da função. A função recebida na resposta foi diferente da esperada pela requisição.
5	ERR_FRAME_DATA_COUNT	A quantidade de dados da resposta foi diferente do esperado
6	ERR_INVALID_PROTOCOL_ID	Protocolo não identificado. O protocolo da resposta é diferente do esperado.
7	ERR_NOT_ECHO	Resposta não é eco da pergunta (FC 5 e 6).
8	ERR_REFERENCE_NUMBER	Número de referência inválido (FC 15 e 16).
9	ERR_INVALID_FRAME_SIZE	Resposta menor do que a esperada.
20	ERR_CONNECTION	Erro durante a fase de conexão.
21	ERR_SEND	Erro durante a fase de transmissão.
22	ERR_RECEIVE	Erro durante a fase de recepção.
40	ERR_CONNECTION_TIMEOUT	Time-out no nível de aplicação durante a conexão.
41	ERR_SEND_TIMEOUT	Time-out no nível de aplicação durante a transmissão.
42	ERR_RECEIVE_TIMEOUT	Time-out no nível de aplicação enquanto aguarda resposta.
128	NO_ERROR	Sem erro desde a inicialização.

Tabela 4-51. Códigos de Erro das relações MODBUS Cliente

ATENÇÃO:
Diferentemente de outras tarefas de uma aplicação, quando for atingida uma marca de depuração na MainTask, a tarefa de uma instância MODBUS Ethernet Cliente, e qualquer outra tarefa MODBUS, irá parar de ser executada no momento em que tentar efetuar uma escrita em uma área de memória. Isto ocorre para manter a consistência dos dados das áreas de memória enquanto a MainTask não estiver em execução.

Configuração do Protocolo MODBUS Ethernet Cliente por Representação Direta (%Q)

Para configurar este protocolo usando Representação Direta (%Q), é necessário executar os seguintes passos:

- Configurar os parâmetros gerais do protocolo MODBUS cliente, como: protocolo e variáveis de representação direta (%Q) para receber os diagnósticos.
- As descrições de cada configuração estão relacionadas a seguir, neste capítulo.

- Adicionar e configurar dispositivos, definindo endereço, variáveis de representação direta (%Q) para desabilitar as relações e porta de comunicação.
- Adicionar e configurar relações MODBUS, especificando o tipo de dado e função MODBUS, time-outs, variáveis de representação direta (%Q) para receber os diagnósticos da relação e outras para receber/escrever os dados, quantidade de dados a comunicar e polling da relação.

As descrições de cada configuração estão relacionadas a seguir, neste capítulo.

Parâmetros Gerais do Protocolo MODBUS Cliente – Configuração por Representação Direta (%Q)

Os parâmetros gerais, encontrados na tela inicial de configuração do protocolo MODBUS (Figura 4-30), são definidos como:

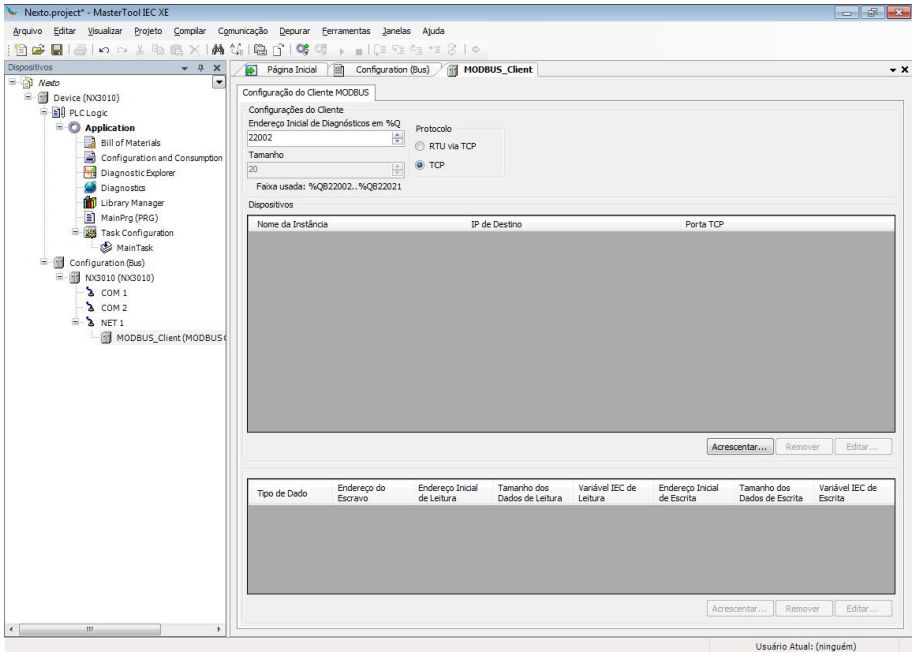

Figura 4-30. Tela de Configuração – MODBUS Cliente

Seleção do protocolo e variáveis de representação direta (%Q) para os diagnósticos:

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Endereço Inicial de Diagnósticos em %Q	Endereço inicial das variáveis de diagnóstico	-	0 a 2147483628
Tamanho	Tamanho da área de diagnósticos	20	Desabilitada para edição
Protocolo	Seleção do protocolo	TCP	RTU via TCP TCP

Tabela 4-52. Configurações Gerais Protocolo – MODBUS Cliente

Notas:

Endereço Inicial de Diagnósticos em %Q: Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Padrão de Fábrica: O padrão de fábrica não pode ser definido para o campo Endereço Inicial de Diagnósticos em %Q, pois a criação de uma instância do protocolo pode ser realizada em qualquer momento no desenvolvimento do aplicativo, fazendo com que o próprio software MasterTool IEC XE aloque um valor, da faixa de variáveis de saída de representação direta (%Q), ainda não utilizado.

Os diagnósticos e comandos do protocolo MODBUS estão descritos na Tabela 4-43.

Configuração dos Dispositivos – configuração por Representação Direta (%Q)

A configuração dos dispositivos clientes, visualizados na Figura 4-31, seguem os seguintes parâmetros:

Figura 4-31. Configurando Cliente MODBUS

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Nome da Instância	Nome da instância	MODBUS_Device	Identificador, segundo a IEC 61131-3
IP de Destino	Endereço IP do servidor	0. 0. 0.1	1.0.0.1 a 223.255.255.255
Porta TCP	Porta TCP	502	2 a 65534
Desabilitação dos Mapeamentos	Endereço inicial utilizado para desabilitar as relações MODBUS	-	Qualquer endereço da área de %Q, limitado conforme UCP utilizada

Tabela 4-53. Configurações do Dispositivo – Cliente MODBUS

Notas:

Nome da Instância: Esse campo é o identificador do dispositivo, o qual é verificado segundo a IEC 61131-3, ou seja, não permite espaços, caracteres especiais e iniciar com caractere numeral. É limitado em 24 caracteres.

Porta TCP: Caso sejam adicionadas várias instâncias do protocolo em uma única interface Ethernet, diferentes portas TCP devem ser selecionadas para cada instância. Algumas portas TCP, entre as possibilidades mencionadas acima, são reservadas e, portanto, não podem ser utilizadas. São elas: 80, 8080, 1217, 1740, 1741, 1742,1743 e 11740.

Desabilitação dos Mapeamentos: Composto por 32 bits, utilizados para desabilitar, individualmente, as 32 relações MODBUS configuradas no espaço Mapeamentos do Dispositivo. A relação é desabilitada quando o bit, correspondente a relação, for igual a 1, caso contrário, o mapeamento está habilitado. Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Padrão de Fábrica: O padrão de fábrica não pode ser definido para o campo Desabilitação dos Mapeamentos, pois a criação de uma instância do protocolo pode ser realizada em qualquer momento no desenvolvimento do aplicativo, fazendo com que o próprio software MasterTool IEC XE aloque um valor, da faixa de variáveis de saída de representação direta (%Q), ainda não utilizado.

Time-out de comunicação: As configurações presentes no botão "Avançado...", relativas à conexão TCP, estão descritas nas notas da seção Configuração dos Dispositivos – Configuração por Mapeamento Simbólico.

Configuração dos Mapeamentos – Configuração por Representação Direta (%Q)

A configuração das relações MODBUS, visualizadas na Figura 4-32 e Figura 4-33, segue os parâmetros descritos na Tabela 4-54:

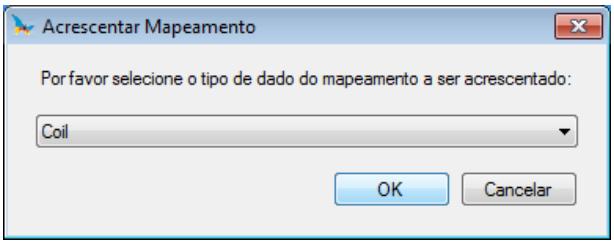

Figura 4-32. Tipo de Dado MODBUS

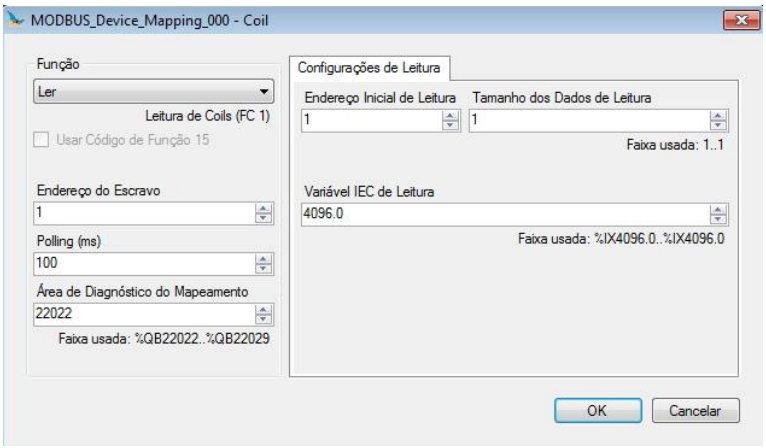

Figura 4-33. Função MODBUS

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Função	Tipo de função MODBUS.	Ler	Ler Escrever Ler/Escrever Máscara de Escrita
Endereço do Escravo	Endereço do escravo MODBUS.	1	0 a 255
Polling (ms)	Período de comunicação (ms).	100	0 a 3600000
Área de Diagnóstico do Mapeamento	Endereço inicial dos diagnósticos da relação MODBUS.	-	0 a 2147483640
Endereço Inicial de Leitura	Endereço inicial dos dados de leitura MODBUS.	1	1 a 65536
Tamanho dos Dados de Leitura	Número de dados de leitura MODBUS.	-	Depende da função utilizada
Variável IEC de Leitura	Endereço inicial das variáveis de leitura (%I).	-	0 a 2147483647
Endereço inicial de Escrita	Endereço inicial dos dados de escrita MODBUS.	1	1 a 65536
Tamanho dos Dados de Escrita	Número de dados de escrita MODBUS.	-	Depende da função utilizada
Variável IEC de Escrita	Endereço inicial das variáveis de escrita (%Q).	-	0 a 2147483647
Máscara de Escrita das Variáveis IEC	Endereço inicial das variáveis para a máscara de escrita (%Q).	-	0 a 2147483644

Tabela 4-54. Mapeamentos do Dispositivo

Notas:

Tabela de Mapeamentos do Dispositivo: O número de configurações e os valores descritos na coluna Possibilidades podem variar de acordo com o tipo de dado e função MODBUS.

Endereço do Escravo: Normalmente, o endereço 0 é utilizado quando o servidor é um Gateway MODBUS TCP ou MODBUS RTU via TCP, e o mesmo transmite a requisição para todos os dispositivos da rede. Quando o endereço 0 é utilizado, o cliente não espera por uma resposta, seu uso serve somente para comandos de escrita. Além disso, de acordo com a Norma MODBUS, a faixa de endereços válidos para escravos é de 0 a 247, sendo os endereços 248 a 255 reservados.

Polling: Este parâmetro indica com que frequência a comunicação definida por esta relação deve ser executada. Ao ser finalizada uma comunicação será aguardado um tempo igual ao polling configurado e, após, será executada uma nova comunicação o mais rápido possível.

Área de Diagnósticos do Mapeamento: Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas. Os diagnósticos da relação MODBUS configurada, são armazenados em quatro bytes e duas words, as quais estão descritos na Tabela 4-50 (n é o valor configurado no campo Área de Diagnóstico do Mapeamento).

Tamanho dos Dados de Leitura e de Escrita: Detalhes do tamanho dos dados suportados por cada função está descrito nas notas da seção Configuração das Requisições – configuração por Mapeamento Simbólico.

Variável IEC de Leitura: Caso o tipo de dado MODBUS seja Coil ou Input Status (bit), o endereço inicial das variáveis IEC de leitura terá o formato por exemplo %IX10.1. Porém, se o tipo de dado MODBUS for Holding Register ou Input Register (16 bits), o endereço inicial das variáveis IEC de leitura terá o formato %IW. Esse campo é limitado pelo tamanho da memória de variáveis de entrada endereçáveis (%I) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Variável IEC de Escrita: Caso o tipo de dado MODBUS seja Coil (bit), o endereço inicial das variáveis IEC de escrita terá o formato por exemplo %QX10.1. Porém, se o tipo de dado MODBUS for Holding Register (16 bits), o endereço inicial das variáveis IEC de escrita terá o formato %QW. Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Máscara de Escrita das Variáveis IEC: A função Máscara de Escrita do Register (FC 22), através de uma lógica entre o valor já escrito e as duas words configuradas neste campo, sendo a word %QW(0) para a máscara AND e a word %QW(2) para a máscara OR; permite ao usuário manipular a word. Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Padrão de Fábrica: O padrão de fábrica não pode ser definido para os campos Área de Diagnósticos do Mapeamento, Variável IEC de Leitura, Variável IEC de Escrita e Máscara de Escrita das Variáveis IEC, pois a criação de uma relação pode ser realizada em qualquer momento no desenvolvimento do aplicativo, fazendo com que o próprio software MasterTool IEC XE aloque um valor, da faixa de variáveis de saída de representação direta (%Q), ainda não utilizado. O padrão de fábrica não pode ser definido para os campos Tamanho de Dados de Leitura e Tamanho de Dados de Escrita, pois eles vão variar de acordo com o tipo de dado MODBUS selecionado.

ATENÇÃO: Diferentemente de outras tarefas de uma aplicação, quando for atingida uma marca de depuração na MainTask, a tarefa de uma instância MODBUS Ethernet Cliente, e qualquer outra tarefa MODBUS, irá parar de ser executada no momento em que tentar efetuar uma escrita em uma área de memória. Isto ocorre para manter a consistência dos dados das áreas de memória enquanto a MainTask não estiver em execução.

MODBUS Ethernet Servidor

Este protocolo está disponível para as UCPs da Série Nexto nos seus canais Ethernet. Ao selecionar esta opção no MasterTool IEC XE, a UCP passa a ser servidor da comunicação MODBUS, permitindo a conexão com dispositivos cliente MODBUS. Este protocolo somente está disponível quando a UCP estiver em modo de execução (Modo Run).

Há dois modos de configuração para este protocolo. Um deles faz uso de Representação Direta (%Q), no qual as variáveis são definidas pelo seu endereço. O outro, chamado Mapeamento Simbólico, tem as variáveis definidas pelo seu nome.

O procedimento para inserir uma instância de protocolo é encontrado com detalhes no Manual de Utilização do MasterTool IEC XE – MU299048.

Configuração do Protocolo MODBUS Ethernet Servidor por Mapeamento Simbólico

Para configurar este protocolo usando Mapeamento Simbólico, é necessário executar os seguintes passos:

- Configurar os parâmetros gerais do protocolo MODBUS servidor, como: porta TCP, seleção de protocolo, filtros de IP para Escrita e para Leitura (disponível no botão de configuração de filtros) e tempos de comunicação (disponível no botão de configurações avançadas do Servidor).
- Adicionar e configurar os mapeamentos MODBUS, especificando o nome da variável, tipo de dados, endereço inicial do dado e tamanho do dado.

As descrições de cada configuração estão relacionadas a seguir, neste capítulo.

Parâmetros Gerais do Protocolo MODBUS Servidor – Configuração por Mapeamento Simbólico

Os parâmetros gerais, encontrados na tela inicial de configuração do protocolo MODBUS (Figura 4-34), são definidos como:

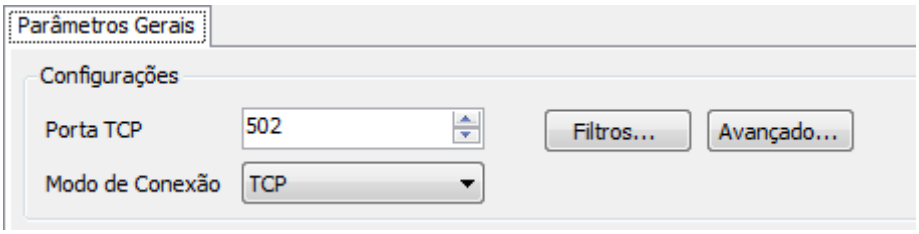

Figura 4-34. Tela de Configuração Parâmetros Gerais MODBUS Servidor

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Porta TCP	Porta TCP.	502	2 a 65534
Modo de Conexão	Seleção do protocolo.	TCP	RTU via TCP TCP

Tabela 4-55. Configurações Gerais MODBUS Servidor

Nota:

Porta TCP: Caso sejam adicionadas várias instâncias do protocolo em uma única interface Ethernet, diferentes portas TCP devem ser selecionadas para cada instância. Algumas portas TCP, entre as possibilidades mencionadas acima, são reservadas e, portanto, não podem ser utilizadas. São elas: 80, 8080, 1217, 1740, 1741, 1742,1743 e 11740.

As configurações presentes no botão “Filtros...”, descritas na Tabela 4-56, são relativas aos filtros de comunicação TCP:

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Filtro de IP para Escrita	Especifica um intervalo de IPs com acesso de escrita nas variáveis declaradas na relação MODBUS.	0. 0. 0. 0	0.0.0.0 a 255.255.255.255
Filtro de Máscara para Escrita	Especifica a máscara de subrede em conjunto com o parâmetro Filtro de IP para Escrita.	0. 0. 0. 0	0.0.0.0 a 255.255.255.255
Filtro de IP para Leitura	Especifica um intervalo de IPs com acesso de leitura nas variáveis declaradas na relação MODBUS.	0. 0. 0. 0	0.0.0.0 a 255.255.255.255
Filtro de Máscara para Leitura	Especifica a máscara de subrede em conjunto com o parâmetro Filtro de IP para Leitura.	0. 0. 0. 0	0.0.0.0 a 255.255.255.255

Tabela 4-56. Filtros de IP

Nota:

Filtros: Os filtros são utilizados para estabelecer um intervalo de endereços IP que têm acesso de escrita ou leitura nas relações MODBUS, sendo individualmente configurados. O critério de permissão é realizado através de uma operação lógica AND entre o Filtro de Máscara para Escrita e o endereço IP do cliente. Caso o resultado seja igual ao Filtro de IP para Escrita, o cliente tem direito de escrita. Por exemplo, se o Filtro de IP para Escrita = 192.168.15.0 e o Filtro de Máscara para Escrita = 255.255.255.0, então somente clientes com endereço IP = 192.168.15.x terão direito de escrita. O mesmo procedimento é aplicado nos parâmetros de Filtro de Leitura para definir os direitos de leitura.

Os tempos de comunicação do protocolo MODBUS Servidor, encontrados no botão “Avançado...” da tela de configuração, estão divididos em: Ciclo da Tarefa e Time-out da Inatividade da Conexão.

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Ciclo da Tarefa (ms)	Tempo para execução da instância dentro do ciclo, sem considerar o tempo de execução da mesma.	50	5 a 100
Time-out da Inatividade da Conexão (s)	Tempo máximo de inatividade entre cliente e servidor antes da conexão ser fechada pelo servidor.	10	10 a 3600

Tabela 4-57. Configurações Avançadas

Notas:

Ciclo da Tarefa: O usuário deverá ter cuidado a alterar esse parâmetro, pois o mesmo interfere diretamente no tempo de resposta, volume de dados por varredura e, principalmente, no balanceamento dos recursos da UCP entre comunicações e outras tarefas.

Time-out da Inatividade da Conexão: Esse parâmetro foi criado para evitar que a quantidade máxima de conexões TCP seja atingida, imaginando que conexões inativas permanecessem abertas pelos mais diversos problemas. Enfim, indica por quanto tempo uma conexão (cliente ou servidora) pode permanecer aberta sem ser utilizada, ou seja, sem trocar mensagens de comunicação. Se o tempo especificado for atingido, a conexão simplesmente é fechada, liberando uma entrada na tabela de conexões.

Diagnósticos MODBUS Servidor – Configuração por Mapeamento Simbólico

Os diagnósticos e comandos do protocolo MODBUS Servidor configurado, seja por mapeamento simbólico ou por representação direta, são armazenados em variáveis do tipo T_DIAG_MODBUS_ETH_SERVER_1 e ainda para o mapeamento por representação direta estão

em 4 bytes e 8 word,as quais estão descritas na Tabela 4-58(n é o valor configurado no campo Endereço Inicial de Diagnósticos em %Q):

Variável de Representação Direta	Variável de diagnósticos do tipo T_DIAG_MODBUS_ETH_SERVER_1 *.		Tamanho	Descrição
Bits de diagnóstico:				
%QX(n).0	tDiag.*	bRunning	BIT	O servidor está em execução.
%QX(n).1		bNotRunning	BIT	O servidor não está em execução (ver bit bInterruptedByCommand).
%QX(n).2		bInterruptedByCommand	BIT	O bit bNotRunning foi habilitado, pois o servidor foi interrompido pelo usuário através de bits de comando.
%QX(n).3		bConfigFailure	BIT	Diagnóstico descontinuado.
%QX(n).4		bRXFailure	BIT	Diagnóstico descontinuado.
%QX(n).5		bTXFailure	BIT	Diagnóstico descontinuado.
%QX(n).6		bModuleFailure	BIT	Diagnóstico descontinuado.
%QX(n).7		bDiag_7_reserved	BIT	Reservado.
%QB(n+1)	byDiag_1_reserved		BYTE	Reservado.
Bits de comando, reiniciados automaticamente:				
%QX(n+2).0	tCommand.*	bStop	BIT	Parar o servidor.
%QX(n+2).1		bRestart	BIT	Reiniciar o servidor.
%QX(n+2).2		bResetCounter	BIT	Reiniciar a estatística dos diagnósticos (contadores).
%QX(n+2).3		bDiag_19_reserved	BIT	Reservado.
%QX(n+2).4		bDiag_20_reserved	BIT	Reservado.
%QX(n+2).5		bDiag_21_reserved	BIT	Reservado.
%QX(n+2).6		bDiag_22_reserved	BIT	Reservado.
%QX(n+2).7		bDiag_23_reserved	BIT	Reservado.
%QB(n+3)	byDiag_03_reserved		BYTE	Reservado.
Estatísticas de comunicação:				
%QW(n+4)	tStat.*	wActiveConnections	WORD	Número de conexões estabelecidas entre cliente e servidor (0 a 64).
%QW(n+6)		wTimeoutClosedConnections	WORD	Contador de conexões, entre cliente e servidor, interrompidas depois de um período de inatividade – time-out (0 a 65535).
%QW(n+8)		wClientClosedConnections	WORD	Contador de conexões interrompidas devido à solicitação do cliente (0 a 65535).
%QW(n+10)		wRXFrames	WORD	Contador de frames Ethernet recebidos pelo servidor, sendo que um frame Ethernet pode conter mais de uma requisição (0 a 65535).
%QW(n+12)		wRXRequests	WORD	Contador de requisições recebidas pelo servidor e respondidas normalmente (0 a 65535).
%QW(n+14)		wTXExceptionResponses	WORD	Contador de requisições recebidas pelo servidor e respondidas com códigos de exceção (0 a 65535). Os códigos de exceção estão relacionados abaixo: 1: o código da função (FC) é legal, porém não suportado. 2: relação não encontrada nestes dados MODBUS. 3: valor ilegal para o endereço.

				128: o mestre/cliente não tem direito de escrita ou leitura. 129: a relação MODBUS está desabilitada.
%QW(n+16)		wRXIllegalRequests	WORD	Contador de requisições ilegais. (0 a 65535).
%QW(n+18)		wDiag_18_reserved	WORD	Reservado.

Tabela 4-58. Diagnósticos MODBUS Servidor

Nota:

Contadores: Todos os contadores dos diagnósticos do MODBUS Ethernet Servidor retornam a zero quando o valor limite 65535 é ultrapassado.

Configuração dos Mapeamentos – configuração por Mapeamento Simbólico

Para a configuração dos mapeamentos MODBUS Servidor, visualizadas na Figura 4-35, segue os parâmetros descritos na Tabela 4-59.

Figura 4-35. Tela de Mapeamentos de dados MODBUS Servidor

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Variável de Valor	Nome da variável simbólica.	-	Nome de uma variável declarada em um programa ou GVL
Tipo de Dado	Tipo de dado MODBUS.	-	Coil Input Status Holding Register Input Register
Endereço Inicial do Dado	Endereço inicial dos dados MODBUS.	-	1 a 65536
Endereço Inicial Absoluto do Dado	Endereço inicial absoluto dos dados MODBUS conforme o seu tipo.	-	-
Tamanho do Dado	Tamanho do dado MODBUS.	-	1 a 65536
Faixa de Dados	Faixa de endereços do dado configurado.	-	-

Tabela 4-59. Configuração dos Mapeamentos MODBUS Ethernet

Notas:

Variável de Valor: Esse campo é utilizado para especificar uma variável simbólica na relação MODBUS.

Tipo do Dado: Esse campo é utilizado para especificar o tipo de dado utilizado na relação MODBUS.

Endereço Inicial do Dado: Endereço inicial do dado de um mapeamento MODBUS.

Endereço Inicial Absoluto do Dado: Endereço inicial absoluto dos dados MODBUS conforme o seu tipo. Por exemplo, o Holding Register com endereço 5 possui endereço absoluto 400005. Este é campo é apenas leitura e está disponível para auxiliar na configuração do Client/Master MODBUS que irá comunicar-se com este dispositivo. Os valores dependem do endereço base (offset) de cada tipo de dado MODBUS e do endereço permitido para cada tipo de dado.

Tamanho do Dado: O valor de Tamanho especifica a quantidade máxima de dados que uma relação MODBUS poderá acessar, a partir do endereço inicial. Sendo assim, para ler uma faixa de endereços contínua, é necessário que todos os endereços estejam declarados em uma única relação. Este campo varia de acordo com o tipo de dado MODBUS configurado.

Faixa de Dados: É um campo somente de leitura e informa a faixa de endereços que está sendo usada por esse mapeamento. Ele é formado pela soma dos campos "Endereço Inicial" e "Tamanho". Não podem haver sobreposições de faixa com outros mapeamentos do mesmo "Tipo de Dado".

ATENÇÃO: Diferentemente de outras tarefas de uma aplicação, quando for atingida uma marca de depuração na MainTask, a tarefa de uma instância MODBUS Ethernet Servidor, e qualquer outra tarefa MODBUS, irá parar de ser executada no momento em que tentar efetuar uma escrita em uma área de memória. Isto ocorre para manter a consistência dos dados das áreas de memória enquanto a MainTask não estiver em execução.

Configuração do Protocolo MODBUS Ethernet Servidor por Representação Direta (%Q)

Para configurar este protocolo usando Representação Direta (%Q), é necessário executar os seguintes passos:

- Configurar os parâmetros gerais do protocolo MODBUS servidor, como: tempos de comunicação, endereço e variáveis de representação direta (%Q) para receber os diagnósticos e controlar as relações.
- Adicionar e configurar relações MODBUS, especificando o tipo de dado MODBUS, variáveis de representação direta (%Q) para receber/escrever os dados e quantidade de dados a comunicar.

As descrições de cada configuração estão relacionadas a seguir, neste capítulo.

Parâmetros Gerais do Protocolo MODBUS Servidor – Configuração por Representação Direta (%Q)

Os parâmetros gerais, encontrados na tela inicial de configuração do protocolo MODBUS (Figura 4-36), são definidos como:

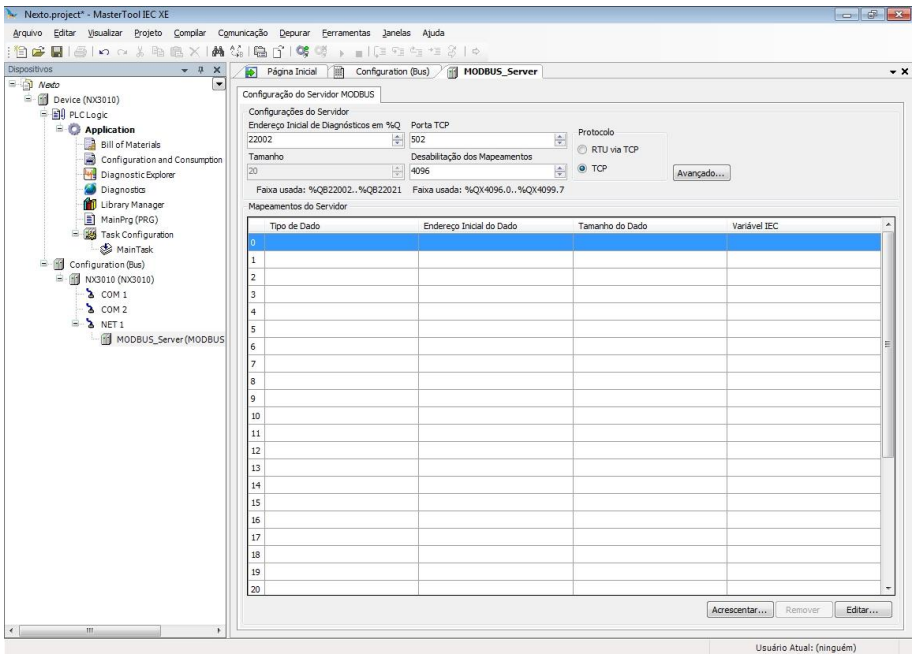

Figura 4-36. Tela de Configuração MODBUS Servidor

- Porta TCP, Protocolo, Variáveis de representação direta (%Q) para controlar as relações e os diagnósticos:

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Endereço Inicial de Diagnósticos em %Q	Endereço inicial das variáveis de diagnóstico.	-	0 a 2147483628
Tamanho	Tamanho da área de diagnósticos.	20	Desabilitada para edição
Porta TCP	Porta TCP.	502	2 a 65534
Desabilitação dos Mapeamentos	Endereço inicial utilizado para desabilitar as relações MODBUS.	-	0 a 2147483644
Protocolo	Seleção do protocolo.	TCP	RTU via TCP TCP

Tabela 4-60. Configurações

Notas:

Endereço Inicial de Diagnósticos em %Q: Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Porta TCP: Caso sejam adicionadas várias instâncias do protocolo em uma única interface Ethernet, diferentes portas TCP devem ser selecionadas para cada instância. Algumas portas TCP, entre as possibilidades mencionadas acima, são reservadas e, portanto, não podem ser utilizadas. São elas: 80, 8080, 1217, 1740, 1741, 1742,1743 e 11740.

Desabilitação dos Mapeamentos: Composto por 32 bits, utilizados para desabilitar, individualmente, as 32 relações MODBUS configuradas no espaço Mapeamentos do Servidor. A relação é desabilitada quando o bit, correspondente a relação, for igual a 1, caso contrário, o mapeamento está habilitado.

Padrão de Fábrica: O padrão de fábrica não pode ser definido para os campos Endereço Inicial de Diagnósticos em %Q e Desabilitação dos Mapeamentos, pois a criação de uma instância do protocolo

pode ser realizada em qualquer momento no desenvolvimento do aplicativo, fazendo com que o próprio software MasterTool IEC XE aloque um valor, da faixa de variáveis de saída de representação direta (%Q), ainda não utilizado.

Os tempos de comunicação do protocolo MODBUS Servidor, encontrados no botão "Avançado..." da tela de configuração, estão divididos em Ciclo da Tarefa (ms) e Time-out da Inatividade da Conexão (s), maiores detalhes estão descritos na seção Parâmetros Gerais do Protocolo MODBUS Servidor – Configuração por Mapeamento Simbólico.

Os diagnósticos e comandos do protocolo MODBUS estão descritoas na Tabela 4-58.

Configuração dos Mapeamentos – Configuração por Representação Direta (%Q)

A configuração das relações MODBUS, visualizada na Figura 4-37 e Figura 4-38, segue os parâmetros descritos na Tabela 4-61:

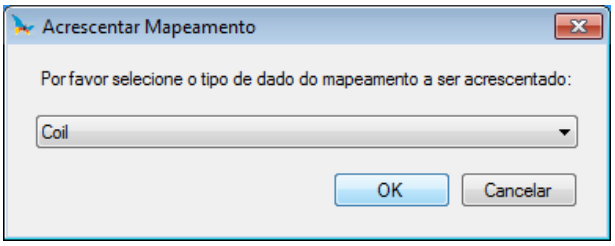

Figura 4-37. Tipo de Dado MODBUS

Figura 4-38. Função MODBUS

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Tipo de Dado	Tipo de dado MODBUS.	Coil	Coil (1 bit) Holding Register (16 bits) Input Status (1 bit) Input Register (16 bits)
Endereço Inicial do Dado	Endereço inicial dos dados MODBUS.	1	1 a 65536
Tamanho do Dado	Número de dados MODBUS.	8	1 a 65536 (Holding Register e Input Register) 8 a 65536 (Coil e Input Status)
Variável IEC	Endereço inicial das variáveis (%Q).	-	0 a 2147473647
Somente Leitura	Somente permite a leitura.	Desabilitada	Habilitada ou Desabilitada

Tabela 4-61. Mapeamentos do Servidor

Notas:

Possibilidades: Os valores descritos na coluna Possibilidades podem variar de acordo com o tipo de dado MODBUS configurado.

Tamanho do Dado: O valor de Tamanho do Dado define a quantidade máxima de dados que uma relação MODBUS poderá acessar, a partir do endereço inicial. Sendo assim, para ler uma faixa de endereços contínua, é necessário que todos os endereços estejam declarados em uma única relação. Este campo varia de acordo com o tipo de dado MODBUS configurado, ou seja, quando selecionado tipo Coil ou Input Status, o campo Tamanho do Dado deve ser um número múltiplo de oito. Também deve-se dar atenção para que o valor máximo não seja superior ao tamanho da memória de saídas endereçáveis e não sejam atribuídos os mesmos valores já utilizados durante a aplicação.

Variável IEC: Caso o tipo de dado MODBUS seja Coil ou Input Status (bit), o endereço inicial das variáveis IEC terá o formato por exemplo %QX10.1. Porém, se o tipo de dado MODBUS for Holding Register ou Input Register (16 bits), o endereço inicial das variáveis IEC terá o formato %QW. Esse campo é limitado pelo tamanho da memória de variáveis de saídas endereçáveis (%Q) de cada UCP, a qual pode ser consultada no capítulo Descrição Técnica – Características Específicas.

Somente Leitura: Quando habilitada, somente permite que o mestre da comunicação leia os dados das variáveis, não permitindo a escrita. Opção válida somente para as funções de escrita.

Padrão de Fábrica: O padrão de fábrica não pode ser definido para o campo Variável IEC, pois a criação de uma instância do protocolo pode ser realizada em qualquer momento no desenvolvimento do aplicativo, fazendo com que o próprio software MasterTool IEC XE aloque um valor, da faixa de variáveis de saída de representação direta (%Q), ainda não utilizado. O padrão de fábrica não pode ser definido para o campo Tamanho do Dado, pois ele vai variar de acordo com o tipo de dado MODBUS selecionado.

As configurações presentes no botão “Filtros...”, descritas na Tabela 4-62, são relativas aos filtros de comunicação TCP:

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Filtro de IP para Escrita	Especifica um intervalo de IPs com acesso de escrita nas variáveis declaradas na relação MODBUS	0. 0. 0. 0	0.0.0.0 a 255.255.255.255
Filtro de Máscara para Escrita	Especifica a máscara de subrede em conjunto com o parâmetro Filtro de IP para Escrita	0. 0. 0. 0	0.0.0.0 a 255.255.255.255
Filtro de IP para Leitura	Especifica um intervalo de IPs com acesso de leitura nas variáveis declaradas na relação MODBUS	0. 0. 0. 0	0.0.0.0 a 255.255.255.255

Filtro de Máscara para Leitura	Especifica a máscara de subrede em conjunto com o parâmetro Filtro de IP para Leitura	0. 0. 0. 0	0.0.0.0 a 255.255.255.255

Tabela 4-62. Filtros de IP MODBUS Servidor

Nota:

Filtros: Os filtros são utilizados para estabelecer um intervalo de endereços IP que têm acesso de escrita ou leitura nas relações MODBUS, sendo individualmente configurados. O critério de permissão é realizado através de uma operação lógica AND entre o Filtro de Máscara para Escrita e o endereço IP do cliente. Caso o resultado seja igual ao Filtro de IP para Escrita, o cliente tem direito de escrita. Por exemplo, se o Filtro de IP para Escrita = 192.168.15.0 e o Filtro de Máscara para Escrita = 255.255.255.0, então somente clientes com endereço IP = 192.168.15.x terão direito de escrita. O mesmo procedimento é aplicado nos parâmetros de Filtro de Leitura para definir os direitos de leitura.

Nas relações definidas anteriormente, o tamanho máximo de dados MODBUS pode ser 65536 (máximo valor configurado no campo Tamanho do Dado). Porém, a requisição que chega ao MODBUS Ethernet Servidor deverá endereçar um subconjunto desse mapeamento e esse grupo deve ter, no máximo, o tamanho de dados que depende do código da função, os quais estão definidos abaixo:

- Leitura de Coils (FC 1): 2000
- Leitura de Input Status (FC 2): 2000
- Leitura de Holding Registers (FC 3): 125
- Leitura de Input Registers (FC 4): 125
- Escrita de Single Coil (FC 5): 1
- Escrita de Single Holding Register (FC 6): 1
- Forçamento de Multiple Coils (FC 15): 1968
- Escrita de Holding Registers (FC 16): 123
- Máscara de Escrita (FC 22): 1
- Leitura/Escrita de Holding Registers (FC 23):
    - Leitura: 121
    - Escrita: 121

ATENÇÃO:
Diferentemente de outras tarefas de uma aplicação, quando for atingida uma marca de depuração na MainTask, a tarefa de uma instância MODBUS Ethernet Servidor, e qualquer outra tarefa MODBUS, irá parar de ser executada no momento em que tentar efetuar uma escrita em uma área de memória. Isto ocorre para manter a consistência dos dados das áreas de memória enquanto a MainTask não estiver em execução.

OPC DA

Para comunicar com as UCPs da Série Nexto é possível utilizar a tecnologia OPC DA (Open Platform Communications Data Access). Esta plataforma de comunicação aberta foi desenvolvida para ser o padrão utilizado nas comunicações industriais. Baseado na arquitetura cliente/servidor, oferece inúmeras vantagens no desenvolvimento de projeto e facilidades na comunicação com os sistemas de automação.

Uma analogia muito comum, utilizada para descrever a tecnologia OPC, é a de uma impressora. Quando corretamente conectada, o computador precisa de um driver para ter a interface com o equipamento. Muito similar, o OPC auxilia na interface entre o sistema de supervisão com os dados de campo no CP.

Quando se trata do desenvolvimento de projetos, configurar a comunicação e trocar informações entre os sistemas é extremamente simples utilizando tecnologia OPC. Utilizando outros drivers,

baseados em endereços, é necessário criar tabelas para relacionar as tags do sistema de supervisão e as variáveis do controlador programável. Quando as áreas de dados são alteradas, no decorrer do desenvolvimento do projeto, é necessário refazer os mapeamentos e novas tabelas com as relações entre as informações do CP com o sistema de Supervisão Controle e Aquisição de Dados (SCADA).

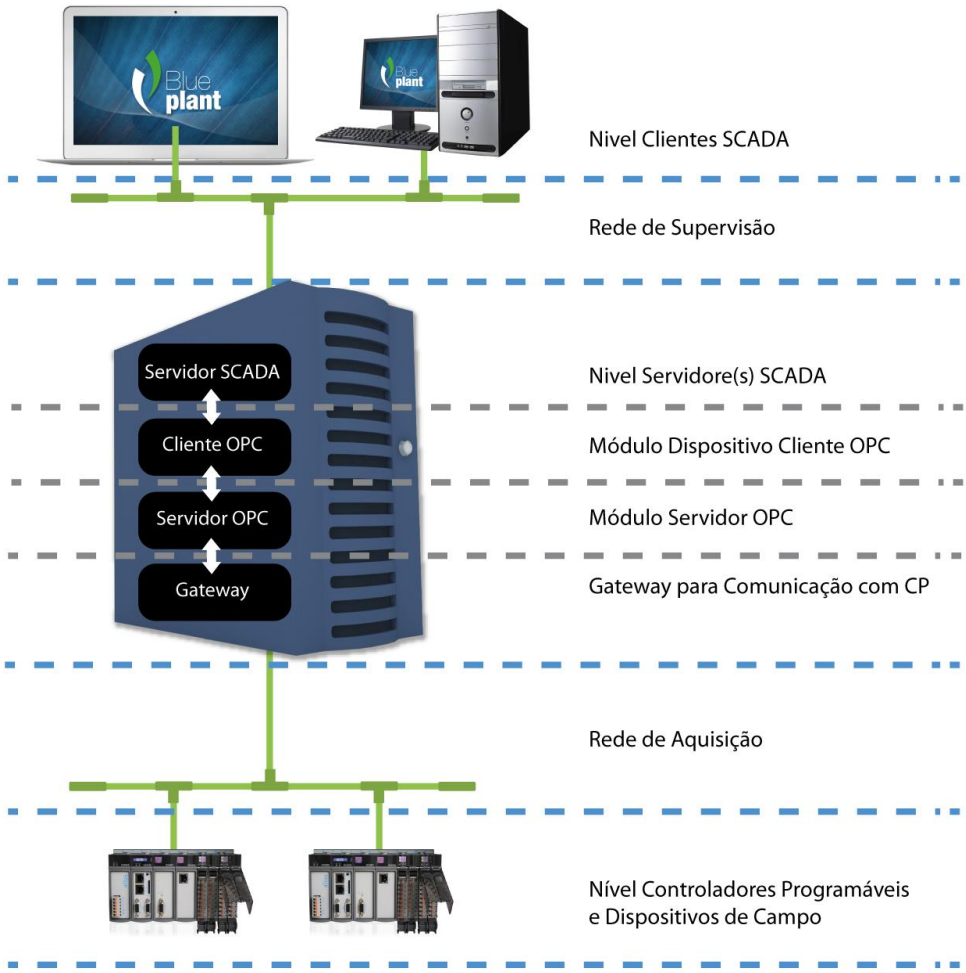

Figura 4-39. Arquitetura OPC

A Figura 4-39 apresenta uma arquitetura para comunicação de sistema SCADA e CPs em projeto de automação. Todos os papéis presentes na comunicação estão explícitos nesta figura independente do local onde estejam executando, eles podem estar em um mesmo equipamento ou em equipamentos diferentes. Cada um dos papéis desta arquitetura é descrito na Tabela 4-63.

Papel	Descrição
Nível Controladores Programáveis e Dispositivos de Campo	Os dispositivos de campo e o CPs são os dispositivos nos quais as informações do estado de operações e controle da planta são armazenados. Os sistemas SCADA acessam as informações nestes dispositivos e armazenam nos servidores SCADA para consulta pelos Clientes SCADA durante a operação da planta.
Rede de Aquisição	A rede de aquisição é a rede na qual trafegam as mensagens para solicitar os dados que são coletados dos dispositivos de campo.
Gateway para Comunicação com CP	Para a comunicação entre o Servidor OPC e os CPs da Série Nexto é utilizado um gateway que permite

	esta comunicação. Sempre é necessário existir um gateway na mesma subrede do CP como descrito no capítulo Configurações de Comunicação, no Manual de Utilização MasterTool IEC XE – MU299048.
Módulos Servidor OPC	O Servidor OPC é um Módulo responsável por receber as requisições OPC DA e traduzi-las para a comunicação com os dispositivos de campo.
Módulo Dispositivo Cliente OPC	O módulo Dispositivo do Cliente OPC é responsável por fazer requisições ao Servidores OPC utilizando o protocolo OPC DA. Os dados coletados por ele são armazenados na base de dados do Servidor SCADA.
Nível Servidor SCADA	O Servidor SCADA é responsável por se conectar ao diversos dispositivos de comunicação e armazenar os dados coletados destes dispositivos em uma base de dados para que possa ser consultado pelos Clientes SCADA.
Rede de Supervisão	A rede de supervisão é a rede pela qual os Clientes SCADA estão conectados aos Servidores SCADA. Em uma topologia na qual não se usa diversos Clientes ou que o servidor e o Cliente estejam instalados em um mesmo equipamento, não existe este tipo de rede.
Nível Clientes SCADA	Os clientes SCADA são responsáveis por solicitar aos servidores SCADA os dados necessários para exibir em uma tela onde é executada a operação de uma planta. Através deles é possível executar leituras e escritas em dados armazenados na base de dados do Servidor SCADA.

Tabela 4-63. Descrição dos Papéis em uma Arquitetura com Servidor OPC

A relação entre as tags dos sistemas de supervisão e os dados do processo nas variáveis do controlador é totalmente transparente. Isso significa que se as áreas de dados são alteradas no decorrer do desenvolvimento do projeto, não há a necessidade de refazer relações entre as informações do CP com o SCADA. Basta utilizar a nova variável disponibilizada pelo CP nos sistemas que requisitam esse dado.

O uso do OPC oferece maior produtividade e conectividade com os sistemas SCADA. Contribui na redução do tempo de desenvolvimento de aplicações e nos custos com manutenção. Possibilita, ainda, inserção de novos dados na comunicação de forma simplificada com maior flexibilidade e interoperabilidade entre os sistemas de automação por ser um padrão aberto.

O Servidor OPC é instalado juntamente com a instalação do MasterTool IEC XE e sua configuração é realizada dentro da ferramenta. Vale salientar que o OPC está disponível somente nas interfaces Ethernet locais das UCPs Nexto. Os módulos de expansão Ethernet não suportam essa funcionalidade.

Criando um Projeto para Comunicação OPC

Diferente das comunicações com drivers como MODBUS e PROFIBUS DP, para configurar a comunicação OPC basta configurar o nó corretamente e indicar quais as variáveis que serão utilizadas na comunicação. Existem duas formas de indicar quais as variáveis de projeto estarão disponíveis no Servidor OPC. Em ambos os casos é necessário adicionar o objeto *Symbol Configuration* à aplicação, caso este não esteja presente. Para adicioná-lo basta clicar com o botão direito do mouse sobre o objeto *Application* e selecionar a opção.

ATENÇÃO:
As variáveis exibidas no objeto *IoConfig_Globals, IoConfig_Application_Mappings e IoConfig_Global_Mappings* são utilizadas internamente para controle de E/S e não devem ser utilizadas pelo usuário.

ATENÇÃO:
Além das variáveis declaradas nas POUs em linguagem SFC são exibidas algumas variáveis criadas implicitamente. Para cada passo criado é criada uma variável do tipo *IecSfc.SFCStepType* onde podem ser monitorados os estados do passo, ou seja, se o mesmo é ativo ou não e o tempo que é ativo conforme define a norma IEC61131-3. Para cada transição é criada uma variável do tipo BOOL que define se a transição é verdadeira ou falsa. Essas variáveis são exibidas no objeto *Symbol Configuration* podendo ser disponibilizadas para acesso pelo Cliente OPC.

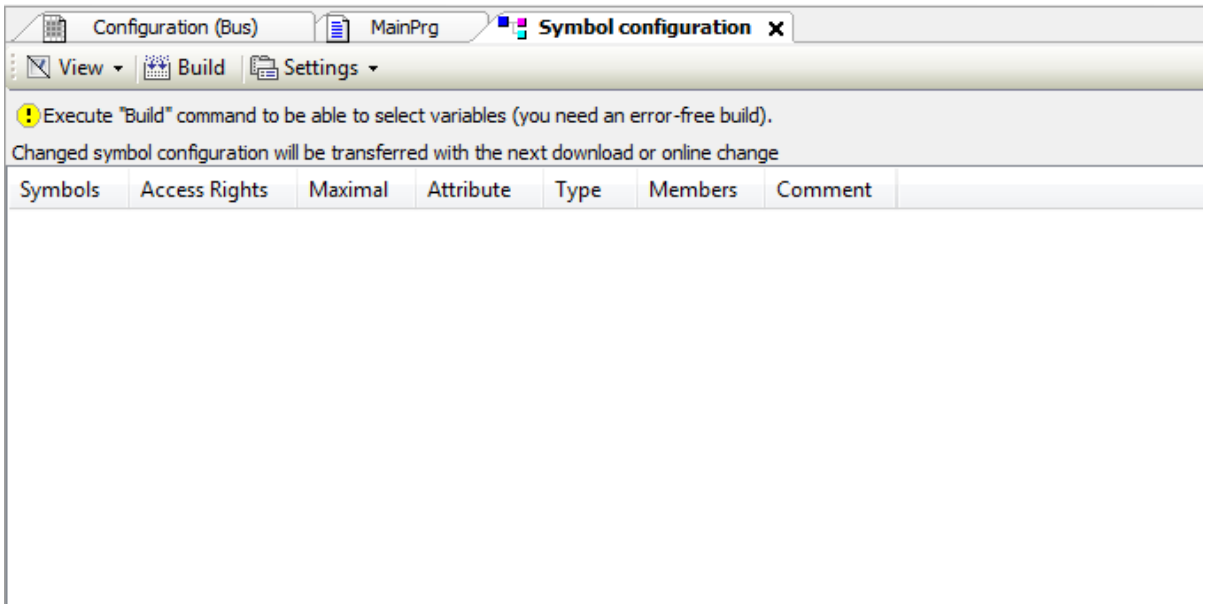

Figura 4-40. Objeto Symbol Configuration

A Tabela 4-64 apresenta a descrição dos campos da tela de configurações dos símbolos no objeto *Symbol Configuration*.

Campo	Descrição
Symbols	Identificador da variável que será disponibilizada para o Servidor OPC.
Access Rights	Indica qual o nível de acesso possível no símbolo declarado. Quando não se utiliza esta coluna, a mesma fica vazia e o nível de acesso é máximo. Caso contrário o nível de acesso pode ser modificado clicando sobre o campo. As opções possíveis são as seguintes: Somente leitura Somente escrita Leitura e escrita
Maximal	Indica o máximo nível de acesso que é possível atribuir a variável. Os símbolos que representam tem o mesmo significado do campo Access Right. Não é possível alterar e é indicado pela presença ou não do {attribute 'symbol'}
Attribute	Indica se está sendo utilizado {attribute 'symbol'} quando declarada a variável. Quando não é utilizado esta coluna fica vazia. Para os casos nos quais se usa o atributo o comportamento é o seguinte: {attribute 'symbol' := 'read'} a coluna exibe {attribute 'symbol' := 'write'} a coluna exibe {attribute 'symbol' := 'readwrite'} a coluna exibe
Type	Tipo de dado da variável declarada.

Members	Quando o tipo de dado for uma Struct é habilitado um botão nesta coluna. Ao clicar no botão é possível selecionar quais elementos da estrutura serão disponibilizados para o Servidor OPC.
Comment	Comentário da variável inserido na POU ou GVL onde a mesma é declarada. Para aparecer como comentário da variável, o comentário deve ser inserido uma linha antes da declaração da variável, no editor quando em modo texto ou na coluna comentário, quando em modo tabular.

Tabela 4-64. Descrição dos Campos da Tela do Objeto Symbol Configuration

Ao executar uma alteração nas configurações do projeto, como adicionar ou remover variáveis, se faz necessário executar o comando de *Build* para atualizar a lista de variáveis. Este comando deve ser executado até que a mensagem presente na Figura 4-40 desapareça. Após isso todas as variáveis disponíveis no projeto, sejam declaradas em POUs, GVLs e diagnósticos, serão exibidas e podem ser selecionadas. As variáveis selecionadas estarão disponíveis no Servidor OPC para acesso pelos Clientes.

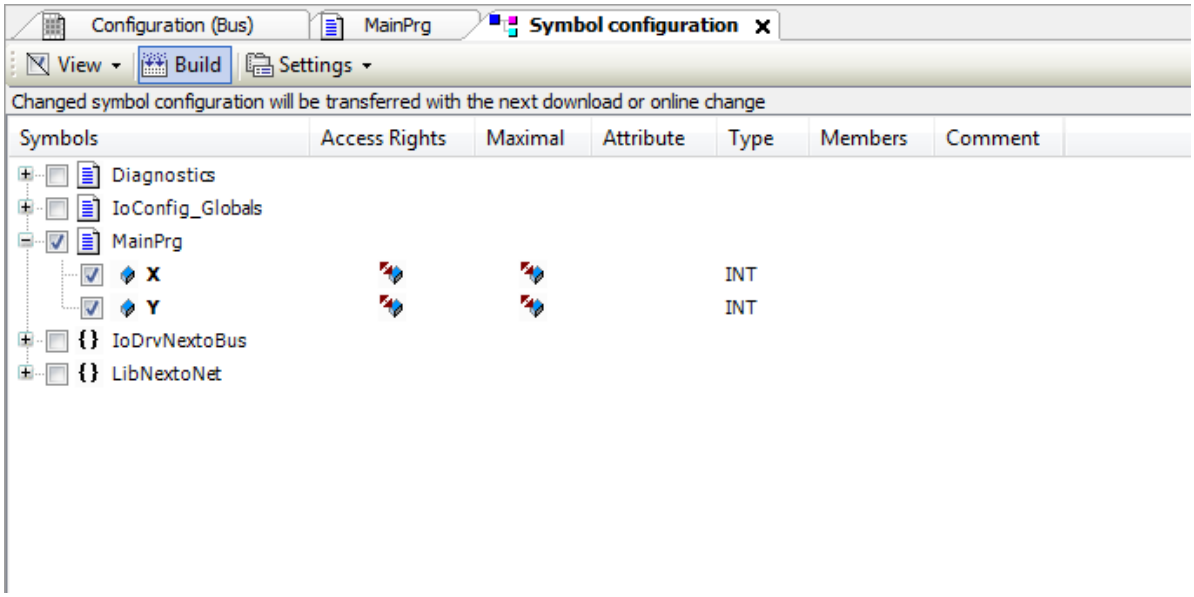

Figura 4-41. Selecionando Variáveis na Symbol Configuration

Após este procedimento o projeto pode ser carregado em um CP e as variáveis selecionadas estarão disponíveis para comunicação com o Servidor OPC. Se a tela do Objeto *Symbol Configuration* estiver aberta e alguma das variáveis, POUs ou GVLs selecionadas for alterada, os nomes destes objetos aparecerão na cor vermelha. As situações nas quais isso acontece é caso a variável seja deletada ou o valor do atributo tenha sido modificado.

Também é possível configurar quais as variáveis estarão disponíveis no Servidor OPC através de um atributo inserido diretamente nas POUs ou GVLs onde as variáveis são declaradas. Quando o atributo {attribute 'symbol'} está presente na declaração das variáveis, podendo estar antes da definição do nome da POU ou GVL, ou para cada variável individualmente, estas são enviadas diretamente para o objeto *Symbol Configuration*, as quais são apresentadas com um símbolo na coluna *Attribute*. Antes de carregar o projeto neste caso é necessário executar o comando de *Build* dentro do objeto *Symbol Configuration*.

As sintaxes válidas para uso do atributo são:

- {attribute 'symbol' := 'none'} – quando o valor do atributo for igual a 'none' as variáveis não serão disponibilizadas para o Servidor OPC e não serão exibidas na tela do objeto *Symbol Configuration*.

- {attribute 'symbol' := 'read'} - quando o valor do atributo for igual a 'read' as variáveis serão disponibilizadas para o Servidor OPC com direito de acesso somente de leitura.
- {attribute 'symbol' := 'write'} - quando o valor do atributo for igual a 'write' as variáveis serão disponibilizadas para o Servidor OPC com direito de acesso somente de escrita.
- {attribute 'symbol' := 'readwrite'} – quando o valor do atributo for igual a 'readwrite' as variáveis serão disponibilizadas para o Servidor OPC com direito de acesso de leitura e escrita.

No exemplo a seguir de declaração de variáveis, a configuração das variáveis A e B permite que um Servidor OPC acesse as mesmas com direito de acesso para leitura e escrita. Em contraponto a variável C não pode ser acessada, enquanto a variável D é acessada com direito de acesso apenas para leitura.

```
{attribute 'symbol' := 'readwrite'}
PROGRAM MainPrg
VAR
A: INT;
B: INT;
{attribute 'symbol' := 'none'}
C: INT;
{attribute 'symbol' := 'read'}
D :INT;
END_VAR
```

Quando uma variável diferente dos tipos básicos é definida, o uso do atributo deve ser feito dentro da declaração desta DUT e não somente no contexto onde a variável é declarada. Por exemplo, no caso de uma instância DUT declarada dentro de uma POU ou GVL que possuem um atributo, este não irá impactar no comportamento dos elementos da instância desta DUT. Será necessário aplicar o mesmo nível de acesso na declaração da DUT.

ATENÇÃO: As configurações dos símbolos que serão disponibilizados ao Servidor OPC são armazenadas dentro do projeto do CP. Ao modificar estas configurações é necessário carregar a aplicação no CP para que seja possível acessar estas variáveis.

ATENÇÃO: Quando uma variável é excluída do projeto e carregada no CP desmarcando a mesma do objeto *Symbol Configuration*, a variável não pode mais ser lida com o Cliente OPC. Se a variável for adicionada ao projeto novamente, com o mesmo nome e o mesmo contexto, e inserida no objeto *Symbol Configuration*, será necessário reinicializar o Cliente OPC para atualizar a referência do endereço da variável, que passa a ser criada em outra área de memória no CP.

Configurando um CP no Servidor OPC

A configuração de um CP é executada dentro do MasterTool IEC XE através da opção disponível no menu *Comunicação*. É necessário que o MasterTool IEC XE seja executado como administrador.

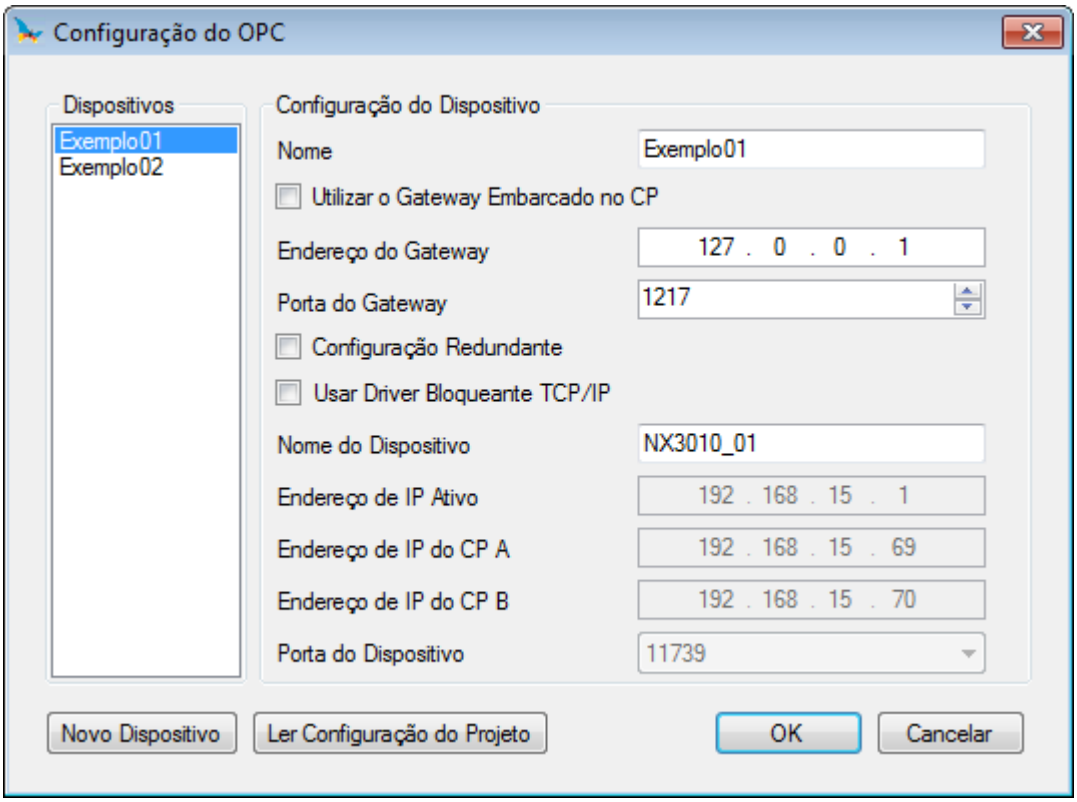

Figura 4-42. Configuração do OPC Server

A *Configuração do Gateway* é a mesma configurada no Gateway utilizado para comunicação entre o MasterTool IEC XE e o CP e descrita em **Configurações de Comunicação**, presente no Manual de Utilização MasterTool IEC XE – MU299048. Se a configuração utilizada for localhost o *Endereço do Gateway* deve ser preenchido com 127.0.0.1. Esta configuração é necessária, pois o Servidor OPC utiliza o mesmo gateway de comunicação e o mesmo protocolo utilizados na comunicação entre CP e MasterTool IEC XE.

Existe a opção *Utilizar o Gateway Embarcado no CP* que pode ser selecionada quando se deseja utilizar o Gateway que fica no próprio CP. Esta opção pode ser empregada para otimizar a comunicação, pois ela evita o excesso de trafego através de uma determinada estação, quando mais de uma estação, com Cliente OPC, esteja conectada ao mesmo CP.

Para a configuração do CP, são possíveis dois tipos de configuração, conforme a seleção do checkbox *Usar Driver Bloqueante TCP/IP*. Quando a opção não está selecionada o nome do CP deve ser colocado no campo *Nome do Dispositivo*. Este é o nome exibido pelo CP selecionado como ativo na tela de *Configurações de Comunicação*.

A outra opção é usar o *Endereço de IP* das Interfaces Ethernet. O mesmo endereço configurado nas telas de configuração deve ser colocado neste campo. Além disso, quando for utilizado este método deve ser colocado o número da porta 11740. A confirmação irá salvar as configurações do Servidor OPC.

Configuração do Dispositivo	Descrição	Padrão de Fábrica	Possibilidades
Nome	Descrição do PLC dentro do arquivo de configuração do Servidor OPC. Este campo pode ter qualquer nome, mas para organização recomenda-se utilizar	'PLC1'	O campo é uma STRING e podem ser colocados caracteres alfanuméricos (letras e números) e o caractere "_". Não é permitido iniciar a STRING com

	o nome do projeto carregado no CP.		números ou "_". Permite até 49 caracteres.
Endereço do Gateway	Endereço IP do computador onde está instalado o Servidor OPC para os casos onde todos os CPs estejam na mesma subrede. Caso exista algum CP em outra subrede deve ser especificado o Gateway utilizado nesta subrede.	127.0.0.1	0.0.0.0 a 255.255.255.255
Porta do Gateway	Porta TCP para a conexão com o Gateway.	1217	2 a 65534
Nome do Dispositivo	É o nome do CP exibido na aba Configuração de comunicação do Device. O nome é a STRING antes do valor em hexadecimal que esta entre [ ]. Somente habilitado quando o checkbox Usar Driver Bloqueante TCP não está selecionado.	'0000'	O campo é uma STRING e podem ser colocados quaisquer caracteres assim como é feito na configuração do nome do CP na aba Configurações de comunicação do Device. Permite até 49 caracteres.
Endereço de IP Ativo	Endereço IP do CP. Somente habilitado quando o checkbox Usar Driver Bloqueante TCP está selecionado. É utilizado somente quando a configuração não for redundante.	192.168.15.1	0.0.0.0 a 255.255.255.255
Endereço de IP CP A	Endereço IP do CPA. Somente habilitado quando a configuração for redundante. É o endereço do CP primário com o qual o servidor irá comunicar caso não exista falha.	192.168.15.69	0.0.0.0 a 255.255.255.255
Endereço de IP CP B	Endereço IP do CPB. Somente habilitado quando a configuração for redundante. É o endereço do CP secundário com o qual o servidor irá comunicar caso ocorra uma falha.	192.168.15.70	0.0.0.0 a 255.255.255.255
Porta do Dispositivo	Porta TCP. Somente habilitado quando o checkbox Usar Driver Bloqueante TCP está selecionando.	11740	11740 ou 11739

Tabela 4-65. Parâmetros de Configuração de Cada CP para o Servidor OPC

Quando um novo CP precisar ser configurado no Servidor OPC basta pressionar o botão *Novo CP* que a configuração será criada. Sempre que a tela de configuração for acessada será exibida uma lista com todos os CPs já configurados no Servidor OPC. As configurações existentes podem ser editadas selecionando o CP na lista *Devices* e editando os parâmetros. As configurações de CPs que não são mais utilizadas podem ser excluídas. O número máximo de CPs configurados em um Servidor OPC é 16.

Caso a arquitetura de automação utilizada preveja que o servidor OPC deve ser executado em um computador onde não é executada a comunicação com o CP via MasterTool IEC XE, a ferramenta deve ser instalada neste computador para permitir a configuração do Servidor OPC da mesma maneira como é feito nas outras situações.

ATENÇÃO:
Para armazenar a configuração do Servidor OPC o MasterTool IEC XE precisar ser executado com direitos de administrador no Sistema Operacional. Dependendo da versão do sistema Operacional este direito deve ser autorizado ao executar o programa. Para essa operação clique com o botão direito sobre o executável do MasterTool IEC XE e escolha a opção Executar como administrador.

ATENÇÃO:
As configurações de um CP no Servidor OPC não são armazenadas no projeto criado no MasterTool IEC XE. Por esta razão podem ser realizadas com um projeto aberto ou fechado. As configurações são armazenadas em um arquivo de configuração onde o Servidor OPC está instalado. Quando alterar as configurações não é necessária carga de aplicação no CP, mas dependendo do Cliente OPC é possível que seja necessário conectar novamente ao Servidor ou carregar as configurações para que os dados sejam atualizados corretamente.

Importando uma Configuração do Projeto

Utilizando o botão *Ler Configuração do Projeto*, conforme a Figura 4-42, é possível atribuir a configuração do projeto aberto à configuração do CP que está em edição. Para que esta opção funcione corretamente deve existir um projeto aberto e deve ser definido um Caminho Ativo conforme descrito em **Configurações de Comunicação**, presente no Manual de Utilização MasterTool IEC XE – MU299048. Caso alguma destas condições não seja atendida será exibida uma mensagem de erro e nenhum dado será modificado.

Quando as condições anteriores são válidas, as configurações do CP recebem os parâmetros do projeto aberto. As informações de *Endereço IP* e *Porta do Gateway* são as configuradas conforme descrito em **Configurações de Comunicação** de acordo com o Caminho Ativo. Já as configurações de Endereço de IP são lidas das configurações da interface Ethernet NET1. A porta para conexão com o CP é sempre atribuída neste caso como 11740.

Configuração com CP no Servidor OPC com Redundância de Conexão

É possível configurar o Servidor OPC para que este opere com redundância de conexão. Desta forma o Servidor OPC irá se comunicar preferencialmente com um CP, mas quando, por alguma razão, não conseguir estabelecer uma comunicação com este CP um segundo CP também configurado será acessado. Esta configuração é especialmente importante para a comunicação de sistemas SCADA com os CPs da Série Nexto que utilizam redundância de Half-Cluster, onde existe um CP em estado ativo executando o processo e outro CP em estado reserva apto a assumir o controle do processo quando ocorrer algum tipo de falha.

A configuração do projeto nestes casos é semelhante a descrita em Criando um Projeto para Comunicação OPC. Contudo, quando um Projeto é criado com Redundância de Half-Cluster e a comunicação com o sistema de supervisão se dará através do Servidor OPC, é necessário selecionar a opção da *Configuração de Redundância OPC* como habilitada durante o Wizard de criação de

projeto do MasterTool IEC XE. Ao habilitar esta opção o projeto criado terá o código necessário para funcionamento da comunicação com redundância de conexão OPC.

No caso redundante, uma variável é declarada dentro da POU, chamada *NonSkippedPrg*. Esta POU é executada em ambos os CPs, independente do estado de redundância. Dentro desta POU é declarada uma variável do tipo BOOL, utilizada para o controle da conexão com o Servidor OPC chamado *OPCRedundancyActive*. Esta variável pode ser acessada de qualquer ponto da aplicação, através de todo o contexto, ou seja, *Application.NonSkippedPrg. OPCRedundancyActive*. Ela é declarada dentro do objeto *Symbol Configuration* com direito apenas de leitura por parte do SCADA. Quando o valor da variável for igual a TRUE os dados são lidos através da conexão com este CP. Desta forma toda vez que ocorre uma troca de estado entre os CPs a variável tem seu estado alterado, permanecendo no estado TRUE no CP que está no estado ativo de redundância.

O código do programa *NonSkippedPrg* é o seguinte, em linguagem ST:

```
PROGRAM NonSkippedPrg
VAR
      {attribute 'symbol' := 'read'}
      OPCRedundancyActive : BOOL;
END_VAR

IF fbRedundancyManagement.m_fbDiagnosticsLocal.eRedState =
REDUNDANCY_STATE.ACTIVE THEN
      OPCRedundancyActive := TRUE;
ELSE
      OPCRedundancyActive := FALSE;
END_IF
```

O código do programa *NonSkippedPrg* pode ser editado tomando-se o cuidado de manter o código acima sem alterações. Este código testa o estado da redundância e preenche uma variável do tipo BOOL chamada *OPCRedundancyActive*, em função deste estado. Caso o CP seja o Ativo, o valor da variável será TRUE, caso contrário será FALSE. Esta variável recebe o attribute ‘symbol’ := ‘read’ para permitir que o Servidor OPC acesse o seu conteúdo e defina de onde a informação deve será lida.

Caso se decida adicionar comunicação OPC após um projeto ter sido criado, é possível configurar o OPC adicionando o código anterior no programa *NonSkippedPrg* e adicionando o objeto *Symbol Configuration* ao projeto.

Para a configuração do CP redundante no Servidor OPC é necessário selecionar a opção *Configuração Redundante* na tela de configuração conforme exibido na Figura 4-42. Quando esta opção é selecionada, será sempre utilizada a opção *Usar Driver Bloqueante TCP/IP*. Além disso, serão habilitados os campos *Endereço IP CPA* e *Endereço IP CPB* conforme descrito na Tabela 4-65. Estes Endereços de IP são os mesmos configurados nas interfaces Ethernet dentro do projeto do CP com redundância de Half-Cluster. Para facilitar a configuração quando um projeto redundante estiver aberto o botão *Ler Configuração do Projeto* pode ser utilizado.

ATENÇÃO: A redundância de conexão do Servidor OPC é realizada por apenas um Servidor. Nos casos em que se deseje uma maior disponibilidade dos dados para os sistemas de supervisão deve ser usada uma arquitetura com Servidores SCADA redundantes. Nestes casos não é necessária nenhuma configuração no Servidor OPC. Consulte as documentações do sistema SCADA para verificar quais as configurações necessárias para o funcionamento da arquitetura redundante.

Variáveis de Status e Qualidade da Comunicação OPC

Para cada um dos CPs criados no Servidor OPC são geradas variáveis de status chamadas de *_CommState* e *_CommStateOK*. A variável *_CommState* indica o estado da comunicação do Servidor OPC com CP. Este estado pode ser interpretado pelo Cliente OPC conforme a Tabela 4-66.

Estado	Valor	Descrição
STATE_TERMINATE	-1	Se a comunicação do Servidor OPC com o Cliente OPC for encerrada este valor será retornado. Quando houver mais de um Cliente OPC conectado ao mesmo tempo o retorno ocorrerá na desconexão do último Cliente conectado. Este status apesar de estar na variável não pode ser visualizado, pois só é alterado quando não existe mais conexão com um cliente.
STATE_PLC_NOT_CONNECTED	0	O CP configurado no Servidor OPC não está conectado. Pode acontecer caso a configuração esteja incorreta (Endereço de IP do CP e ou Gateway errados) ou o CP não esteja disponível naquele momento.
STATE_PLC_CONNECTED	1	O CP configurado no Servidor OPC está conectado. Este é um estado transitório durante conexão.
STATE_NO_SYMBOLS	2	Não existem símbolos (variáveis) disponíveis no CP configurado no OPC Server. Pode acontecer na situação de não existirem símbolos ou ainda quando não houver um projeto carregado no CP.
STATE_SYMBOLS_LOADED	3	Finalizado o processo de leitura dos símbolos (variáveis) do CP configurado no Servidor OPC. Este é um estado transitório durante conexão.
STATE_RUNNING	4	Após leitura dos símbolos (variáveis) o Servidor OPC está executando a atualização periódica dos valores dos símbolos disponíveis em cada CP configurado.
STATE_DISCONNECT	5	Ocorreu desconexão com o CP configurado no servidor OPC.
STATE_NO_CONFIGURATION	6	Quando a configuração do OPC (armazenada em um arquivo OPCServer.ini) possuir uma sintaxe errada este será o status da variável. De maneira geral, este comportamento não é observado pois o MasterTool IEC XE mantém esta configuração integra.

Tabela 4-66. Descrição dos estados da comunicação do Servidor OPC com o CP

A variável *_CommStateOK* é uma variável do tipo BOOL que indica se a comunicação está funcionando entre o Servidor OPC e o CP. Quando o valor é TRUE, indica que a comunicação está funcionando corretamente. Se o valor for FALSE não é possível comunicar por alguma razão com o CP.

Além de monitorar o estado da comunicação, o Cliente OPC pode acessar informações da qualidade de comunicação. Os bits de qualidade formam um byte. Eles estão divididos em três grupos de bits: Qualidade, Substatus e Limite. Os bits estão distribuídos da seguinte forma QQSSSSLL, onde QQ representa os bits de Qualidade, SSSS os bits de Substatus e LL os bits de Limite. Neste caso os bits QQ são os mais significativos no byte, enquanto os bits LL são os menos significativos.

QQ	Valor dos Bits	Definição	Descrição
0	00SSSSLL	Bad	O valor lido não pode ser usado por que existe alguma problema na conexão. É possível monitorar o valor de _CommState diagnosticar o problema.
1	01SSSSLL	Uncertain	A qualidade não pode ser definida e pode ser apresentada no campo Substatus.
2	10SSSSLL	NA	Este valor é reservado e não é utilizado pelo padrão OPC.
3	11SSSSLL	Good	O qualidade é boa e o valor lido pode ser usado.

Tabela 4-67. Descrição do valor da Qualidade OPC

A Tabela 4-67 apresenta os valores possíveis de qualidade. O Servidor OPC retorna apenas valor com Qualidade Good e Bad. Um Cliente OPC pode manter a qualidade como Uncertain em caso de alguma falha na qual ele não consiga uma conexão com o Servidor. No caso de monitoração dos 8

bits de qualidade diretamente do Servidor OPC os campos Substatus e Limite serão nulos e a Qualidade Good será representada com o valor 192 e a qualidade Bad com o valor 0.

Limites da Comunicação com Servidor OPC

Para comunicação com o Servidor OPC existem algumas limitações que devem ser respeitadas para o correto funcionamento da aplicação. Não devem ser configurados para estarem disponíveis no Servidor OPC mais de 20.000 variáveis em cada CP. Portanto, 20.000 variáveis é o limite máximo a se comunicar com um único CP.

Além disso quando configurar as variáveis a serem disponibilizadas no Servidor OPC, a quantidade de variáveis declarada em cada POU ou GVL não deve ultrapassar o limite de 5.000. A Tabela 4-68 apresenta os limites de configuração do Servidor OPC.

Número máximo de variáveis comunicando com um único CP	20.000
Número máximo de variáveis declaradas em uma única POU ou GVL	5.000
Número máximo de CPs em um Servidor OPC	16
Número máximo de conexão simultâneas de Servidor OPC em um mesmo CP	8

Tabela 4-68. Limites da Comunicação com o Servidor OPC

ATENÇÃO:
O número máximo de conexões simultâneas de Servidor OPC em um mesmo CP é compartilhado com as conexões realizados com o MasterTool IEC XE. Ou seja, a soma de conexões com Servidor OPC e MasterTool IEC XE não deve ultrapassar o número máximo definido na Tabela 4-68.

A comunicação entre o Servidor OPC e o CP utiliza o mesmo protocolo utilizado para comunicação do MasterTool IEC XE com o CP. Este protocolo só está disponível para as interfaces Ethernet das UCPs da Série Nexto, não sendo possível estabelecer este tipo de comunicação com módulos de expansão Ethernet.

Quando uma comunicação é estabelecida entre o Servidor OPC e o CP estes dois elementos iniciam uma série de transações que visam resolver o endereço de cada variável declarada, otimizando a comunicação em regime de leitura de dados. Além disso, nesta fase também são resolvidas as classificações dos grupos de comunicação usados por alguns Clientes com o intuito de otimizar a comunicação. Este processo inicial demanda algum tempo e depende da quantidade de variáveis mapeadas.

O tempo aproximado desta fase inicial com 5.000 variáveis é de aproximadamente 2 min. No casos em que mais variáveis são utilizadas este tempo pode ser de até 4 min, dependendo do tipo de dado e das configurações da rede.

Acessando Dados Através de um Cliente OPC

Após a configuração do Servidor OPC os dados disponíveis em todos os CPs podem ser acessados via um Cliente OPC. Na configuração do Cliente OPC deve ser selecionado o nome do Servidor OPC correto. Neste caso o nome é CoDeSys.OPC.DA. A Figura 4-43 exibe a seleção do servidor no driver cliente do software SCADA BluePlant.

ATENÇÃO:
Da mesma forma que o MasterTool IEC XE algumas ferramentas precisam ser executadas com direitos de administrador no Sistema Operacional para o correto funcionamento do Cliente OPC. Dependendo da versão do Sistema Operacional este direito deve ser autorizado ao executar o programa. Para essa operação clique com o botão direito sobre o executável da ferramenta e escolha a opção Executar como administrador.

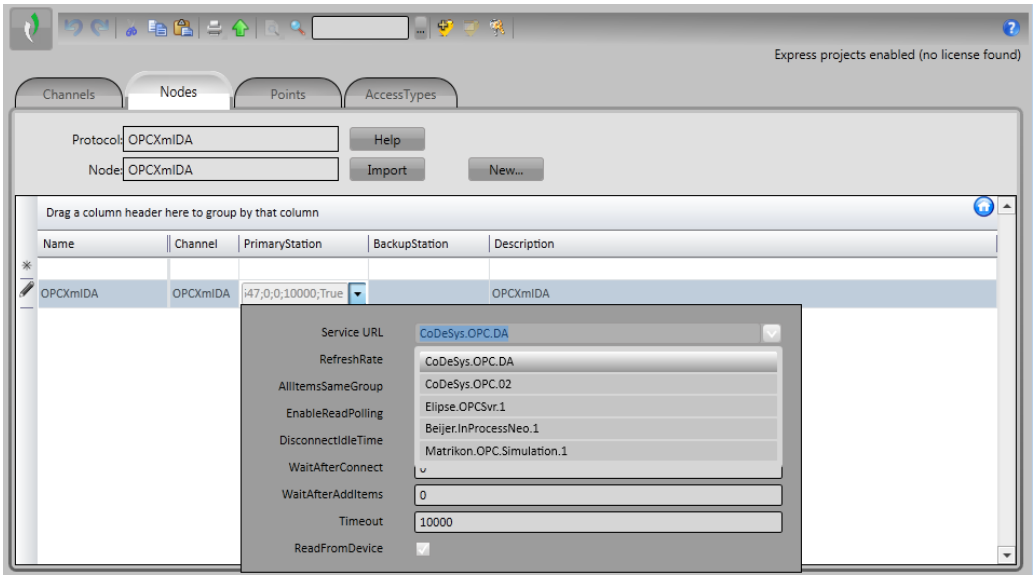

Figura 4-43. Selecionado Servidor OPC na Configuração do Cliente

Nos casos em que o servidor se encontra remotamente pode ser necessário adicionar o caminho da rede ou o endereço de IP do computador onde se encontra o servidor instalado. Nestes casos existem duas opções de configuração. A primeira delas é configurar diretamente para isso sendo necessário liberar os Serviços de COM/DCOM do Windows. Contudo, uma forma mais simples é utilizar uma ferramenta de *tunneller* que abstrai as configurações de COM/DCOM, além de possibilitar uma comunicação mais segura entre o Cliente e o Servidor. Para mais informações sobre este tipo de ferramenta consultar https://www.matrikonopc.com/products/opc-data-management/opc-tunneller.aspx.

Uma vez que o Cliente se conecta no Servidor podem ser usados comandos de importação de TAGs. Estes comandos consultam informações declaradas no CP, retornando uma lista com todos os símbolos disponibilizados por este.

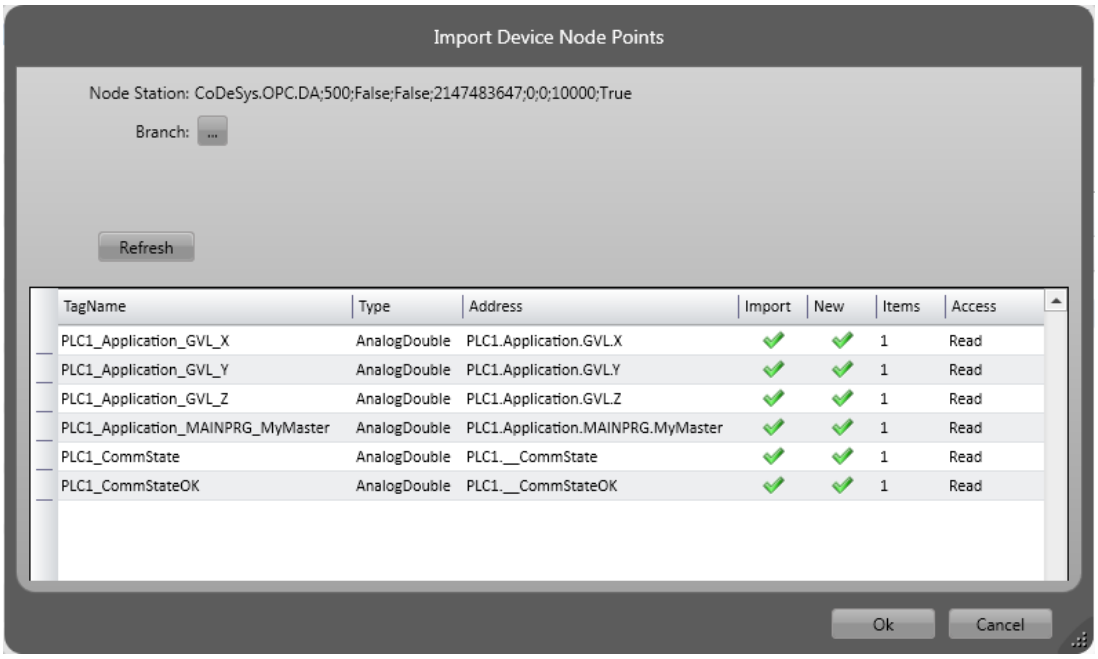

Figura 4-44. Lista de Símbolos Consultados pelo Cliente OPC

A lista de variáveis selecionadas será incluída na lista de comunicações do Cliente e podem ser utilizadas, por exemplo, em telas de um sistema SCADA.

ATENÇÃO: O modo de simulação do software MasterTool IEC XE pode ser utilizado para testes da comunicação OPC. As informações sobre como configurá-lo estão presentes na seção **Testando a Comunicação OPC com o Uso do Simulador**, do Manual de Utilização MasterTool IEC XE – MU299048.

EtherCAT

EtherCAT (Ethernet Control Automation Technology) é um protocolo com arquitetura mestre-escravo de alto desempenho, para Ethernet determinística, que permite performance em tempo real pois atualiza 1000 E/S distribuídas em 30 µS ou 100 eixos de servomotores a cada 100 µS usando par trançado ou cabos de fibra óptica. Além disto, este possui topologia flexível, permitindo ligações em linha, árvore e/ou estrela.

Um frame Ethernet pode ser processado em tempo real em vez de ser recebido, interpretado e copiado como dados do processo em cada conexão. O FMMU (Fieldbus Memory Management Unit) em cada nó Escravo lê os dados que lhe são endereçados ao mesmo tempo em que o telegrama é encaminhado para o próximo dispositivo. De uma maneira similar, os dados de entrada são inseridos enquanto o telegrama é passado. Devido a isso, os frames são atrasados apenas por alguns nanosegundos. Acessos nos terminais de Ethernet podem ser feitos em qualquer ordem, porque a sequência de dados é independente da ordem física. É possível executar comunicação Broadcast, Multicast e comunicação entre os Escravos.

O protocolo EtherCAT permite uma sincronização com precisão, que é necessária, por exemplo, em aplicações onde vários eixos realizam movimentos coordenados simultaneamente, esta pode ser realizada através de um ajuste exato do *Distributed Clock*. Há também a possibilidade de configuração de dispositivos que, em contraste com a comunicação síncrona, têm um elevado grau de tolerância dentro do sistema de comunicação.

A configuração de módulos EtherCAT inicialmente é determinada pelos *Arquivos de Descrição de Dispositivo* dos dispositivos Mestre e Escravo utilizados e pode ser modificada pelo usuário nas caixas de diálogo do *Editor de Configuração*. No entanto, para aplicações convencionais e com o

objetivo de ter uma manipulação tão fácil quanto possível, a configuração em larga escala pode ser automatizada, escolhendo o modo *Autoconfiguração* em Parâmetros do Mestre.

Observe a possibilidade de modificação dos parâmetros de configuração do Mestre e Escravo também no modo operacional, através das instâncias de Mestre e Escravo, de acordo com a disponibilidade do dispositivo em questão.

Instalando e Inserindo Dispositivos EtherCAT

A fim de ser possível inserir e configurar dispositivos EtherCAT como objetos na árvore de dispositivos, os dispositivos Escravos devem ser instalados.

O dispositivo Mestre é instalado automaticamente pela instalação padrão do MasterTool IEC XE. O Mestre EtherCAT define quais Escravos podem ser inseridos.

Para instalar os dispositivos Escravos deve ser aberto o diálogo *Repositório de Dispositivos*, utilizar o filtro *Arquivo de Configuração da Descrição do Dispositivo EtherCAT XML (*.xml)* e selecionar os arquivos de descrição de dispositivo (*EtherCAT XML Device Description* / *ESI EtherCAT Slave Information*), fornecido com o hardware. As descrições para os Escravos estão disponíveis como arquivos XML (tipo de arquivo: *.xml).

Um Mestre EtherCAT pode ser adicionado à *Árvore de Dispositivos* através do comando *Acrescentar Dispositivo*, através do menu de contexto dos conectores NET da CPU.

Abaixo de um mestre, um ou mais escravos podem ser inseridos, selecionando um Mestre EtherCAT e executando o comando *Acrescentar Dispositivo* (menu de contexto do Mestre EtherCAT) ou executando o comando Procurar Dispositivos.

Além das topologias de linha e de árvore o MasterTool IEC XE também suporta a topologia em estrela de EtherCAT. Para a configuração de uma topologia em estrela EtherCAT junções especiais EtherCAT são necessárias (no exemplo: Beckhoff EK1122). Uma estrela EtherCAT modular pode ser realizada utilizando várias junções. Dispositivos individuais ou uma seção de linha EtherCAT completa podem ser conectados nas portas de junção. Uma junção EtherCAT está marcada com o ícone . O exemplo de *Árvore de Dispositivos* na Figura 4-45 mostra diferentes possibilidades.

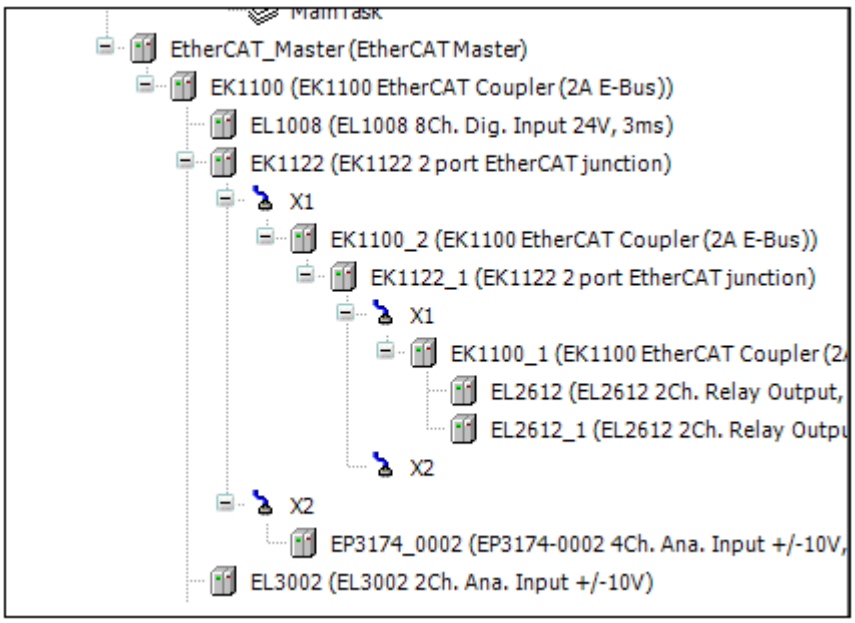
MainTask
EtherCAT_Master (EtherCAT Master)
EK1100 (EK1100 EtherCAT Coupler (2A E-Bus))
EL1008 (EL1008 8Ch. Dig. Input 24V, 3ms)
EK1122 (EK1122 2 port EtherCAT junction)
X1
EK1100_2 (EK1100 EtherCAT Coupler (2A E-Bus))
EK1122_1 (EK1122 2 port EtherCAT junction)
X1
EK1100_1 (EK1100 EtherCAT Coupler (2
EL2612 (EL2612 2Ch. Relay Output,
EL2612_1 (EL2612 2Ch. Relay Outpu
X2
X2
EP3174_0002 (EP3174-0002 4Ch. Ana. Input +/-10V,
EL3002 (EL3002 2Ch. Ana. Input +/-10V)

Figura 4-45. Exemplo de configuração EtherCAT

ATENÇÃO:
- Permitido apenas uma instância de Mestre EtherCAT por projeto.
- Permitido apenas em CPs modelo NX3020 e NX3030.
- Permitido apenas em conectores NET da CPs.
- Não pode ser usado quando as NETs estiverem configuradas como redundantes.
- Não pode ser usado quando o projeto tiver redundância de cluster.
- Não podem ser instanciados outros drivers na mesma porta NET que o Mestre EtherCAT.
- Permitido no máximo 128 escravos EtherCAT por projeto.

Procurar Dispositivos

O comando *Procurar Dispositivos*, disponível no menu de contexto do Mestre EtherCAT, executa uma busca pelos dispositivos Escravos instalados fisicamente na rede EtherCAT do CP atualmente conectado. Isso significa que com este comando é possível detectar e visualizar os componentes de hardware na janela apresentada na figura abaixo, permitindo que o usuário possa mapeá-los diretamente na *Árvore de Dispositivos* do projeto.

É importante salientar que, quando o comando *Procurar Dispositivos* é selecionado, uma conexão com o CP será estabelecida automaticamente antes da busca iniciar e será encerrada quando a busca terminar. Assim, para que este comando seja executado pela primeira vez, a conexão do Gateway deve ser configurada e deve-se fazer um download do projeto com um Mestre EtherCAT configurado no CP. No caso de futuras adições de dispositivos Escravos, para executar este comando é necessário que a rede EtherCAT esteja parada, para isso, colocar em TRUE o bit *bStopBus* presente nas variáveis de Diagnóstico do Mestre EtherCAT.

Quando o comando for executado, o campo *Dispositivos Mapeados* vai conter uma lista de todos os dispositivos e módulos encontrados durante a última verificação. Para adicioná-los ao projeto, basta clicar no botão *Copiar Tudo Para o Projeto*. É possível, também, executar uma comparação entre os dispositivos encontrados na busca com os presentes no projeto, selecionando a caixa *Mostrar Diferenças Para o Projeto*.

Se você adicionar um módulo Mestre EtherCAT ao projeto e utilizar o comando Procurar Dispositivos você terá uma lista de todos os Escravos EtherCAT disponíveis. Aparecerão entradas escritas em negrito, caso exista mais de um dispositivo com a mesma descrição. Com um duplo clique sobre esta entrada uma lista será aberta e o dispositivo desejado pode ser selecionado.

Após concluir as modificações na configuração da rede EtherCAT, é necessário executar um novo download do projeto, para que as modificações sejam aplicadas.

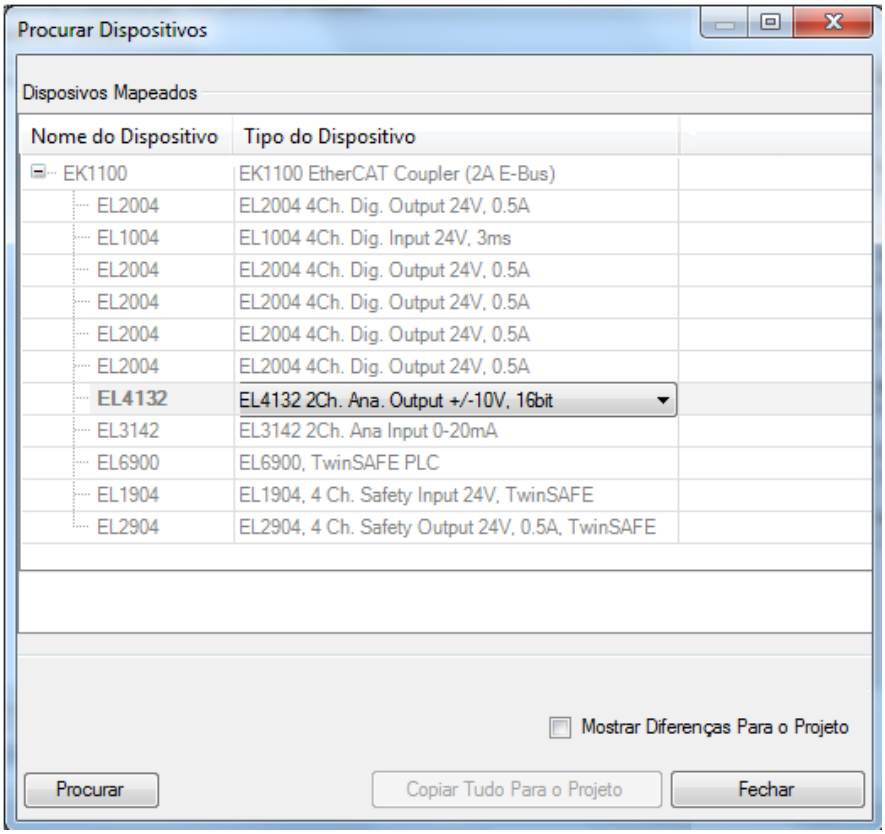

Figura 4-46. Diálogo Procurar Dispositivos para EtherCAT

Diagnóstico de Variáveis

Através da inserção de um Mestre EtherCAT e Escravo é adicionada uma variável de diagnóstico para o dispositivo na GVL *Diagnostics*. Esta variável fornece informações sobre o estado do dispositivo. Existem dois tipos de variáveis, uma para o Mestre EtherCAT e outra para os escravos EtherCAT. Cada variável tem informações específicas sobre o dispositivo. Os diagnósticos e comandos fornecidos estão descritos nas tabelas a seguir.

Variável DG_EtherCAT_Master.*	Tipo	Valores Possíveis	Descrição
tDiag.bRunning	BOOL	FALSE ou TRUE	Se esta variável é TRUE, a transferência de todos os parâmetros de configuração foi finalizada com sucesso, sem erros encontrados e o barramento não foi parado pelo comando de usuário. A comunicação está em execução.
tDiag.bError	BOOL	FALSE ou TRUE	Esta variável será TRUE se um erro é detectado durante a inicialização da pilha EtherCAT ou se durante a operação, a comunicação com os escravos for interrompida se mais nenhuma mensagem pode ser recebida (por exemplo devido a uma falha no cabo). A causa do erro pode ser descoberta com o auxílio da lista de logs via STRING de erro.
tDiag. eLastErrorCode	ETC_LASTERROR		As informações a respeito dos valores possíveis para este diagnóstico, bem como sua descrição são encontrados na Tabela 4-70.
tDiag.bDistributedClockInSync	BOOL	FALSE ou TRUE	Se DC é usado, o CLP será sincronizado com o primeiro escravo EtherCAT cuja configuração DC está ativa. Esta variável é TRUE logo após esta sincronização ser finalizada com sucesso. Este sinal, por exemplo, pode ser usado para inicializar blocos funcionais de SoftMotion, em caso de compatibilidade com o dispositivo, depois que o

			CLP está em modo sincronizado, caso contrário pulos na posição podem ocorrer. Na inicialização do CLP a variável é FALSE e mudará para TRUE depois de alguns segundos. Se a sincronização for perdida devido a qualquer falha, a variável irá para FALSE.
tDiag.bReserved_00	BOOL	-	Espaço reservado.
tDiag.bReserved_01	BOOL	-	Espaço reservado.
byReserved_00	BYTE	-	Espaço reservado.
tCommand.bRestart	BOOL	FALSE ou TRUE	Na borda de subida o mestre será reiniciado completamente. Todos os parâmetros de configuração serão recarregados.
tCommand.bStopBus	BOOL	FALSE ou TRUE	Quando esta variável é TRUE, a comunicação é parada. Pacotes EtherCAT não serão mais enviados. Na maioria dos dispositivos uma reinicialização é necessária, por que eles estão em estado de erro.
tCommand. wDCInSyncWindow	BOOL	0..65535	Janela de tempo para bDistributedClockInSync. O jitter tem que estar dentro desta janela para a variável bDistributedClockInSync permanecer em TRUE. Valor padrão: 50 microssegundos.
tCommand. bySlaveUpdatedbyCycle	BOOL	0..128	Este valor define o número de escravos que são lidos por ciclo para preencher as variáveis de diagnósticos dos escravos. Valor 0 significa que nenhum diagnóstico de escravo será atualizado.
tCommand. bReserved_00	BOOL	-	Espaço reservado.
tCommand. bReserved_01	BOOL	-	Espaço reservado.
byReserved_01	BYTE	-	Espaço reservado.

Tabela 4-69. Diagnósticos e Comandos do Mestre EtherCAT

Código	Enumerável	Descrição
00	NO_ERROR	Sem erro, executando.
01	NO_COMM	Sem comunicação. Mais de 100 pacotes não foram recebidos. Possível falha no cabo do mestre.
02	WRONG_WORKING_COUNTER	Working counter para processdata está errado. Um ou mais escravos não estão operacional ou faltando e o working counter esperado não é encontrado.
03	DC_TIME_ZERO	Tempo DC do escravo é sempre zero -> talvez conectores IN e OUT estão errados e nenhum tempo pode ser lido da referência de tempo.
04	OPEN_FIRSTADAPTER_FAILED	Primeiro adaptador de rede não pode ser aberto.
05	OPEN_SECONDADAPTER_FAILED	Secundo adaptador de rede não pode ser aberto.
06	ADAPTER_MISMATCH	Segundo adaptador de rede usa o MAC-ID como o primeiro interface.
07	NO_SLAVES_FOUND	Erro na inicialização dos escravos: possivelmente há escravos faltando ou sem comunicação.
08	VENDOR_ID_WRONG	VendorID não é igual.
09	PRODUCT_ID_WRONG	ProductID não é igual.
10	NUMBER_DEVICE_MISMATCH	Leitura de ProductID ou VendorID sem sucesso, mais escravos na configuração do que na arquitetura real.
11	SDO_WRITE_ERROR	Erro na escrita SDO no início do processo.
12	SDO_TIMEOUT	Timeout do SDO na inicialização do processo.
13	EMERGENCY_RECEIVED	Emergência recebida do dispositivo.
14	IDN_WRITE_ERROR	Erro na escrita IDN na inicialização do processo.
15	IDN_TIMEOUT	Timeout do IDN na inicialização do processo.

Tabela 4-70. Códigos de Erro do Mestre EtherCAT

Variável DG_Escravo.*	Tipo	Valores Possíveis	Descrição
tDiag.wState	ETC_SLAVE_STATE	ETC_SLAVE_BOOT=3 ETC_SLAVE_INIT=1 ETC_SLAVE_PREOPERATIONAL=2 ETC_SLAVE_SAVEOPERATIONAL=4 ETC_SLAVE_OPERATIONAL=8	Estado atual do escravo.
tDiag. dwVendorID	BOOL	Faixa DWORD	Depois da inicialização da pilha EtherCAT, esta variável retorna o vendorID lido do escravo.
tDiag. dwProductID	ETC_LASTERROR	Faixa DWORD	Depois da inicialização da pilha EtherCAT, esta variável retorna o productID lido do escravo.
tDiag. dwRevisionID	BOOL	Faixa DWORD	Depois da inicialização da pilha EtherCAT, esta variável retorna o revisionID lido do escravo.
tDiag. tLastEmergency.	ETC_CO_Emergency		Se uma mensagem é recebida então esta informação é armazenada dentro do escravo e poderá ser lida na aplicação por esta variável. Em adicional também uma mensagem de log é adicionada. Mais informações a respeito deste diagnóstico são encontradas na Tabela 4-72.
tDiag.bReserved_01	BOOL	-	Espaço reservado.
byReserved_00	BYTE	-	Espaço reservado.

Tabela 4-71. Diagnósticos do Escravo EtherCAT

Variável DG_Escravo.tDiag.tLastEmergency.*	Tipo	Código em Hexa	Descrição
wErrorCode	WORD	00XX	Erro de Reset ou Sem erro.
		10XX	Erro genérico.
		20XX	Corrente.
		21XX	Corrente, lado de dentro do dispositivo.
		22XX	Corrente dentro do dispositivo.
		23XX	Corrente, lado de fora do dispositivo.
		30XX	Tensão.
		31XX	Principais Tensões.
		32XX	Tensão dentro do dispositivo.
		33XX	Tensão de saída.
		40XX	Temperatura.
		41XX	Temperatura Ambiente.
		42XX	Temperatura do dispositivo.
		50XX	Hardware do dispositivo.
		60XX	Software do dispositivo.
		61XX	Software Interno.
		62XX	Software do usuário.
		63XX	Conjunto de Dados.
		70XX	Módulos Adicionais.
		80XX	Monitoração.
		81XX	Comunicação.
		82XX	Erro de protocolo.
		8210	PDO não processa devido a erro de tamanho.

		8220	Tamanho do PDO excedido.
		90XX	Erro Externo.
		A000	Transição de PRE-OPERATIONAL para SAFE-OPERATIONAL sem sucesso.
		A001	Transição de SAFE-OPERATIONAL para OPERATIONAL sem sucesso.
		F0XX	Funções adicionais.
		FFXX	Dispositivo Específico.
byErrorRegister	BYTE	Faixa BYTE	Registrador de erro.
tDiag.tLastEmergency.abyErrorField	ARRAY[0..5] OF BYTE	0000-9FFF	Campo de erro específico para Fabricante.
		A000-EFFF	Dados de diagnósticos.
		F000-FFFF	Campo de erro específico para Fabricante.

Tabela 4-72. Conteúdo de ETC_CO_Emergency

Configuração do Mestre EtherCAT

A seguir são listadas as opções para a execução da configuração de um Mestre EtherCAT, tal como definido no *Arquivo de Descrição do Dispositivo.*

Parâmetros do Mestre

A seguir são apresentados os parâmetros gerais encontrados na tela inicial de configuração do Mestre EtherCAT, Figura 4-47.

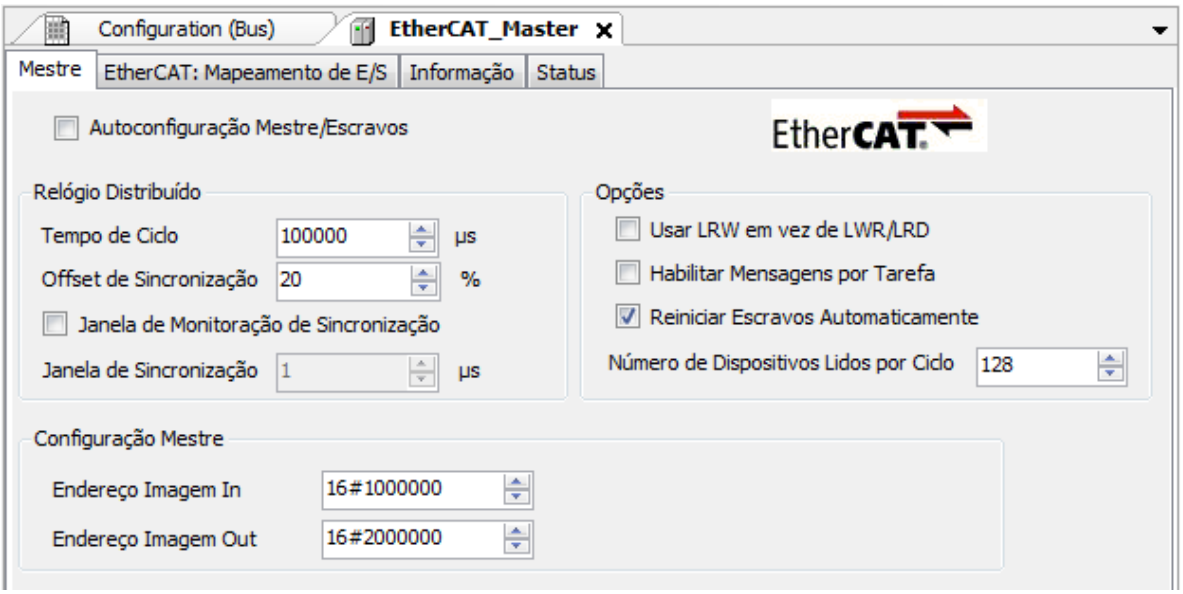

Figura 4-47. Diálogo de Configuração do Mestre EtherCAT

Configuração do Dispositivo	Descrição	Padrão de Fábrica	Possibilidades
Autoconfiguração Mestre/Escravos	Habilita a configuração automática do Mestre e dos Escravos.	Marcado	Marcado Desmarcado
Tempo de Ciclo [µs]	Configura o período de tempo em que um novo telegrama de dados deve ser enviado ao barramento.	100000	2000 a 1000000
Offset de Sincronização [%]	Ajusta o deslocamento da interrupção de sincronização do Escravo EtherCAT para o ciclo do	20	-50 a 50

	CP.		
Janela de Monitoração de Sincronização	Se ativada esta opção permite monitorar a sincronização dos Escravos.	Desmarcado	Marcado Desmarcado
Janela de Sincronização [µs]	Tempo para a Janela de Monitoração de Sincronização.	1	1 a 32768
Usar LRW em vez de LWR/LRD	Habilitação dos comandos combinados de leitura e escrita.	Desmarcado	Marcado Desmarcado
Habilitar Mensagens por Tarefa	Se ativada, os comandos leitura e escrita, que estão lidando com mensagens de entrada e saída poderão ser feitos em diferentes tarefas.	Desmarcado	Marcado Desmarcado
Reiniciar Escravos Automaticamente	Reinicia os escravos quando a comunicação for abortada.	Marcado	Marcado Desmarcado
Número de Dispositivos Lidos por Ciclo	Define o número de Escravos que são lidos por ciclo para preencher as variáveis de diagnóstico. O valor '0' significa que nenhum diagnóstico de Escravo será atualizado.	128	0 a 128
Endereço Imagem In	Primeiro endereço do primeiro Escravo para os dados de entrada.	16#1000000	16#1 a 16#1F000000
Endereço Imagem Out	Primeiro endereço lógico do primeiro Escravo para os dados de saída.	16#2000000	16#1 a 16#1F000000

Tabela 4-73. Configurações do Mestre EtherCAT

Notas:

Autoconfiguração Mestre/Escravos: Se esta opção estiver ativada, a maior parte da configuração do Mestre e Escravo será feita automaticamente, baseando-se nos arquivos de descrição e cálculos implícitos. Neste caso, o diálogo de configurações *FMMU* / *Sync* não estarão disponíveis. Se esta estiver desmarcada as opções *Endereço Imagem In* e *Endereço Imagem Out* estarão disponíveis para o usuário.

ATENÇÃO:
O modo Autoconfiguração é ativado por padrão e, geralmente, suficiente e fortemente recomendado para aplicações padrão. Se a opção estiver desativada, todas as definições de configuração para o Mestre e o Escravo(s) terão de ser feitas manualmente e um conhecimento especializado é necessário. Para a configuração de uma comunicação Escravo-para-Escravo a opção Autoconfiguração tem de ser desativada.

Tempo de Ciclo: Período de tempo após o qual, um novo telegrama de dados deve ser enviado ao barramento. Se a funcionalidade *Distributed Clock* estiver ativada, o valor deste parâmetro será transferido para os clocks dos Escravos. Assim, uma sincronização precisa de troca de dados pode ser alcançada, o que particularmente é importante nos casos em que os processos distribuídos especialmente exigem ações simultâneas. Então, uma base de tempo muito precisa para toda a rede, com um jitter significativamente menor do que um microssegundo, pode ser alcançada.

Offset de Sincronização: Este valor permite ajustar o deslocamento da interrupção de sincronização do Escravo EtherCAT para o ciclo do CP. Normalmente, o ciclo de tarefa do CP começa 20% mais

tarde do que a sincronização de interrupção dos Escravos. Isto significa que a tarefa do CP pode ser atrasada por 80% do tempo de ciclo e nenhuma mensagem será perdida.

Janela de Sincronização: Se a sincronização de todos os Escravos está dentro desta janela de tempo o diagnóstico do Mestre EtherCAT *bDistributedClockInSync* será definido como TRUE, caso contrário será FALSE.

Quando o *Distributed Clock* é utilizado, é altamente recomendado usar uma tarefa dedicada com alta prioridade como *Tarefa Cíclica de Barramento* do mestre EtherCAT. Desta maneira, é necessário utilizar Perfis de Projeto que possibilitem a criação de novas tarefas, criar uma tarefa cíclica com prioridade 0 (tarefa de tempo real), e atribuir a *Tarefa Cíclica de Barramento* do mestre para esta nova tarefa na aba Mapeamento de E/S do Mestre EtherCAT. O usuário também poderá alterar o valor da variável *wDCInSyncWindow*, configurando qual o jitter máximo permitido na sincronização entre mestre e escravos.

Usar LRW em vez de LWR/LRD: A ativação desta opção habilita a comunicação Escravo-para-Escravo, pois, ao invés de comandos separados de leitura (LRD) e escrita (LWR), comandos combinados leitura/escrita (LRW) serão utilizados.

Reiniciar Escravos Automaticamente: Habilitando esta opção o Mestre reiniciará os Escravos assim que a comunicação for abortada, portando, a variável *bError* nos diagnósticos do Mestre EtherCAT na GVL *Diagnostics* não irá para TRUE.

Endereço Imagem In e Endereço Imagem Out: Estas definições só poderão ser editadas se o modo *Autoconfiguração* estiver desativado, caso contrário isto será feito automaticamente e estes parâmetros serão invisíveis.

Mensagem de Diagnóstico: Mostra algumas informações ou mensagens de erro da pilha. As mensagens também são registradas na guia de *Log* do CP, acessada pelo ícone *Device*, na *Árvore de Dispositivos*. Esta opção só fica visível quando o Mestre EtherCAT está em modo Online.

Carga do Barramento: Este valor exibe a carga do barramento no adaptador de rede, esta opção só fica visível quando o Mestre EtherCAT está em modo Online.

Mapeamento de E/S

Esta guia do editor de configuração do Mestre EtherCAT oferece a possibilidade de atribuir as variáveis de projeto para respectivas entradas e saídas do EtherCAT. Assim, as variáveis do Mestre EtherCAT podem ser controladas pela *Aplicação de Usuário*.

Além disso, podemos escolher a opção de *Tarefa Cíclica de Barramento* através da lista de seleção, onde podemos alterar a tarefa que se deseja utilizar. Todas as tarefas adicionadas ao projeto estarão presentes nesta lista. Essa tarefa serve para efetuar as operações do Mestre EtherCAT. A *MainTask* é a opção padrão deste campo.

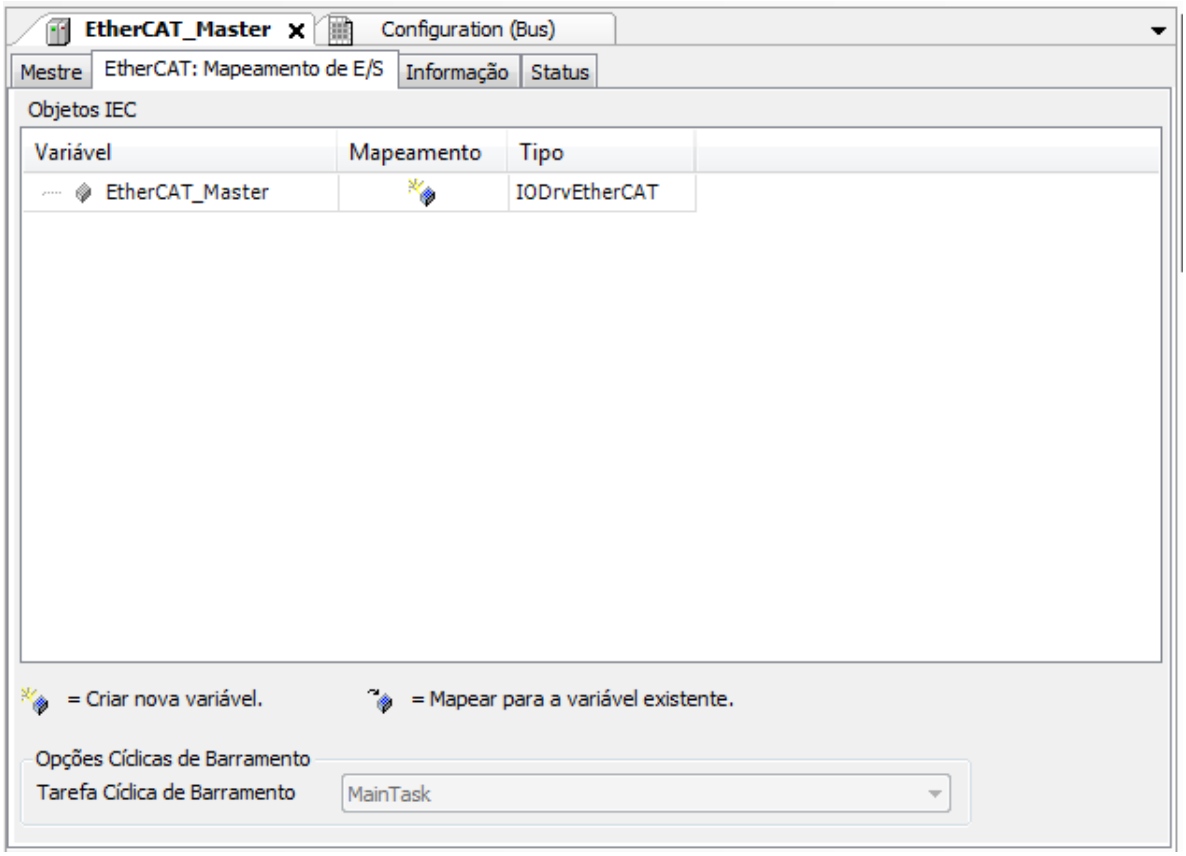

Figura 4-48. Diálogo Mapeamento de E/S

Guias Status e Informação

A guia *Status* do editor de configuração do Mestre EtherCAT fornece informações de estado (por exemplo, 'Executando', 'Parado') e mensagens de diagnóstico específicas do dispositivo e do sistema de barramento interno.

O diálogo *Informação*, presente no editor de configuração do Mestre EtherCAT, exibe, caso disponíveis, as seguintes informações gerais para o módulo: Nome, Fornecedor, Tipo, Número de Versão, Categorias, Número de Ordem, Descrição, Imagem.

Configuração do Escravo EtherCAT

A seguir são listadas as principais opções de configuração para um Escravo EtherCAT, tal como definido no *Arquivo de Descrição do Dispositivo*.

Parâmetros do Escravo

A seguir são apresentados os parâmetros gerais encontrados na tela inicial de configuração do Escravo EtherCAT. Estes campos estarão disponíveis somente se o modo *Autoconfiguração* (Mestre) não estiver ativado.

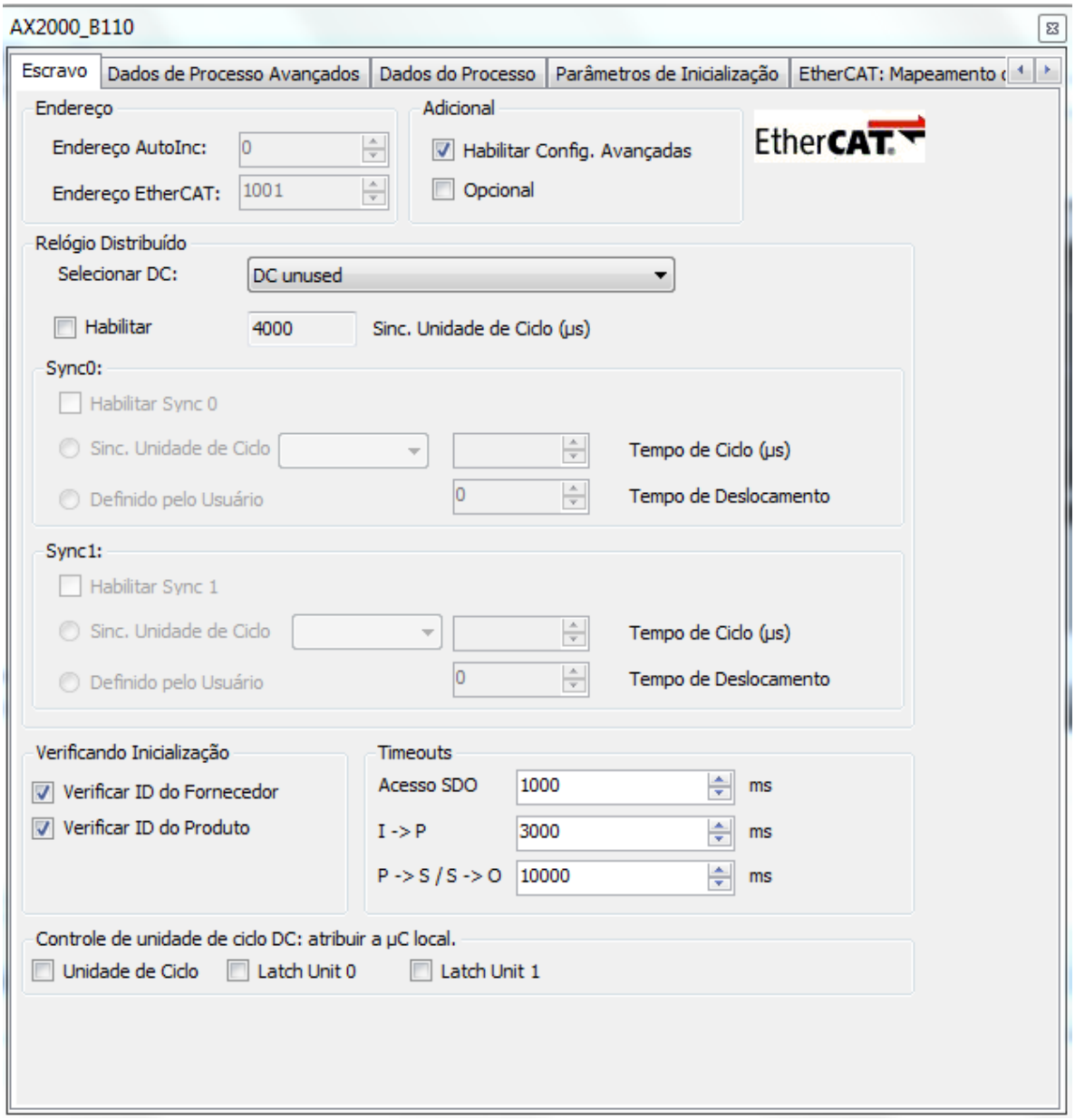

Figura 4-49. Diálogo de Configuração do Escravo EtherCAT

Configuração do Dispositivo	Descrição	Padrão de Fábrica	Possibilidades
Endereço AutoInc	Endereço Autoincremental (16-bit), definido pela posição do Escravo na rede.	-	-65535 a 0
Endereço EtherCAT	Endereço final do Escravo, atribuído pelo Mestre durante a inicialização. Este endereço é independente da posição na rede.	-	1 a 65535
Habilitar Config. Avançadas	Habilita as opções de configuração avançadas do Escravo.	Desmarcado	Marcado Desmarcado
Opcional	Declara o Escravo como Opcional.	Desmarcado	Marcado Desmarcado
Selecionar DC	Apresenta todas as configurações, para		

	Distributed Clocks, fornecidas pelo arquivo de descrição do dispositivo.		
Habilitar Relógio Distribuído	Habilita as opções de configuração do Distributed Clock.	Desmarcado	Marcado Desmarcado
Sinc. Unidade de Ciclo [µs]	Apresenta o Tempo de Ciclo configurado no Mestre.	100000	2000 a 1000000
Habilitar Sync 0	Habilita as configurações da unidade de sincronização Sync 0.	Desmarcado	Marcado Desmarcado
Sinc. Unidade de Ciclo Sync 0	Selecionando esta opção o fator escolhido será multiplicado pelo valor em Sinc. Unidade de Ciclo.	Desmarcado	Marcado Desmarcado
Definido pelo Usuário Sync 0	Se marcado, o tempo desejado em microssegundos pode ser inscrito no campo Tempo de Ciclo (µs).	Desmarcado	Marcado Desmarcado
Tempo de Ciclo [µs] Sync 0	Mostra o tempo de ciclo atualmente definido.	100000	1 a 2147483647
Tempo de Deslocamento Sync 0	Tempo entre os eventos de sincronização e as Outputs Valid ou Input Latch.	0	-2147483648 a 2147483647
Habilitar Sync 1	Habilita as configurações da unidade de sincronização Sync 0.	Desmarcado	Marcado Desmarcado
Sinc. Unidade de Ciclo Sync 1	Selecionando esta opção o fator escolhido será multiplicado pelo valor em Tempo de Ciclo (µs).Sync 0.	Desmarcado	Marcado Desmarcado
Definido pelo Usuário Sync 1	Se marcado, o tempo desejado em microssegundos pode ser inscrito no campo Tempo de Ciclo (µs).	Desmarcado	Marcado Desmarcado
Tempo de Ciclo [µs] Sync 1	Mostra o tempo de ciclo atualmente definido.	100000	1 a 2147483647
Tempo de Deslocamento Sync 1	Tempo entre os eventos de sincronização e as Outputs Valid ou Input Latch.	0	-2147483648 a 2147483647
Verificar ID do Fornecedor	Se desmarcado, desabilita a Verificação do ID do Fornecedor.	Marcado	Marcado Desmarcado
Verificar ID do Produto	Se desmarcado, desabilita a Verificação do ID do Produto.	Marcado	Marcado Desmarcado
Acesso SDO	Configura uma referência para a verificação do tempo Acesso SDO.	6000	0 a 100000
I -> P	Configura uma referência para a verificação do tempo entre a troca do estado Init para Pré-Operacional.	3000	0 a 100000
P -> S/S -> O	Configura uma referência para a verificação do tempo entre a troca dos estados Pré-Operacional para Segurança Operacional e de Segurança Operacional	10000	0 a 100000

	para Operacional.		
Unidade de Ciclo	Atribuí a Unidade de Ciclo ao microprocessador local.	Desmarcado	Marcado Desmarcado
Latch Unit 0	Atribuí a Latch Unit 0 ao microprocessador local.	Desmarcado	Marcado Desmarcado
Latch Unit 1	Atribuí a Latch Unit 1 ao microprocessador local.	Desmarcado	Marcado Desmarcado

Tabela 4-74. Configurações do Escravo EtherCAT

Notas:

Endereço AutoInc: Este endereço é usado apenas durante a inicialização, quando o Mestre está atribuindo os endereços EtherCAT para os Escravos. Quando para este propósito, o primeiro telegrama percorre os Escravos, cada Escravo de leitura rápida aumenta seu Endereço AutoInc por 1. O Escravo com o endereço 0, finalmente, vai receber os dados.

Opcional: Se um Escravo é declarado como *Opcional* nenhuma mensagem de erro será criada caso o dispositivo não exista no sistema de barramento. Se essa opção for ativada, um endereço de estação deve ser armazenado no dispositivo Escravo. Assim, um endereço *Pseudônimo de Estação* devem ser definido e gravado na EEPROM. Esta opção só está disponível se a opção *Autoconfiguração Mestre / Escravo* estiver ativada nas configurações do Mestre EtherCAT, e se esta função for suportada pelo Escravo EtherCAT.

Habilitar Relógio Distribuído: Se a funcionalidade *Distributed Clock* for ativada, o tempo de ciclo de troca de dados, exibido no campo *Unidade de Ciclo Sync(µs)* será determinado pelo *Tempo de Ciclo* do Mestre. Assim, o clock Mestre pode sincronizar a troca de dados dentro da rede. As definições para manusear a(s) unidade(s) de sincronização dependem do Escravo.

Habilitar Sync 0: Se essa opção for ativada, a unidade de sincronização *Sync0* é utilizada. Uma unidade de sincronização descreve um conjunto de dados do processo que são trocados de forma síncrona.

Sinc. Unidade de Ciclo Sync 0: Se esta opção estiver ativada, o *Tempo de Ciclo* do Mestre, multiplicado pelo fator escolhido será utilizado como tempo de ciclo de sincronização para o Escravo. O campo *Tempo de Ciclo(µs)* mostra o tempo de ciclo atualmente definido.

Tempo de Deslocamento: O Tempo de deslocamento descreve o tempo entre os eventos de sincronização (*Sync0*, *Sync1*) e as *Outputs Valid* ou *Input Latch*. Valor gravável, se o escravo suporta transferência de *Outputs Valid* ou *Input Latch.*

Habilitar Sync 1: Se essa opção for ativada, a unidade de sincronização *Sync1* é utilizada. Uma unidade de sincronização descreve um conjunto de dados do processo que são trocados de forma síncrona.

Sinc. Unidade de Ciclo Sync1: Se essa opção for ativada, o *Tempo de Ciclo(µs)* do grupo *Sync0*, multiplicado pelo fator escolhido será utilizado como tempo de ciclo de sincronização para o Escravo. O campo *Tempo de Ciclo(µs)* mostra o tempo de ciclo definido atualmente.

Estado Atual: O estado atual do Escravo é exibido. Valores possíveis: *Init*, *Pré Operacional*, *Segurança Operacional*, *Operacional*. Se o Estado é *Operacional*, a configuração do Escravo foi encerrada corretamente.

Verificar ID do Fornecedor e Verificar ID do Produto: Por padrão na inicialização do sistema o ID do Fornecedor e/ou o ID do Produto serão verificados contra as configurações atuais. Se for detectada alguma diferença, o barramento será interrompido e nenhuma ação adicional será executada. Isso serve para evitar o download de uma configuração errada. Esta opções tem a finalidade de desativar esta verificação.

Acesso SDO: Por padrão não há tempo limite definido para a ação de envio de uma lista SDO ao iniciar o sistema. Porém, se houver a necessidade de verificar se esta ação excede certo tempo, este deve ser especificado neste campo.

I -> P: Por padrão não há tempo limite definido para a ação de troca do modo *Init* para o modo *Pré Operacional*. Porém, se houver a necessidade de verificar se esta ação excede certo tempo, este deve ser especificado neste campo.

P -> S / S-> O: Por padrão não há tempo limite definido para a ação de troca dos modos *Pré Operacional* para *Segurança Operacional* e respectivamente de *Segurança Operacional* para *Operacional*. Porém, se houver a necessidade de verificar se esta ação excede certo tempo, este deve ser especificado neste campo.

Controle da Unidade de Ciclo DC: Escolha a(s) opção(ões) desejada(s), relativa às funções de *Distributed Clock*, a fim de definir o que deve ser atribuído ao microprocessador local. O controle é feito no registro 0x980 do Escravo EtherCAT. As configurações possíveis são: *Unidade Cíclica, Unidade Latch 0, Unidade Latch 1*.

Pseudônimo de Estação: Esta definição só é visível se a opção *Opcional* estiver ativada ou se o dispositivo Escravo suportar endereços de alias (definido no *Arquivo de Descrição do Dispositivo*).

Habilitar: Se a definição de *Opcional* não estiver ativada, essa configuração pode ser ativada se for suportada pela descrição do dispositivo Escravo. Ele permite a atribuição direta de um endereço de alias, a fim de obter o endereço dos Escravos independente de sua posição dentro do barramento. Se a opção *Opcional* estiver ativada, esta caixa de seleção estará desativada.

Escrever na EEPROM: Este comando só é visível no modo online. Ele permite escrever o endereço definido na EEPROM do Escravo. Se não for suportado pelo Escravo, este comando não terá nenhum efeito e o dispositivo não irá funcionar como *Escravo Opcional*.

Endereço Atual: Este campo, visível somente no modo online, exibe o endereço real do Escravo. Ele pode ser utilizado para verificar o sucesso do comando *Escrever na EEPROM*.

FMMU/Sync

Este diálogo só será fornecido em uma guia de um editor de configuração Escravo EtherCAT, se o modo *Autoconfiguração* no Mestre estiver desativado. Ele mostra os *FMMUs* e o *Gerenciador Sync* do Escravo, conforme definido pelo *Arquivo de Descrição do Dispositivo*. Essas configurações podem ser retrabalhadas, por exemplo, para configurar uma comunicação Escravo-para-Escravo.

ATENÇÃO: Essas são as Configurações Avançadas, que geralmente não são necessárias para aplicações padrão.

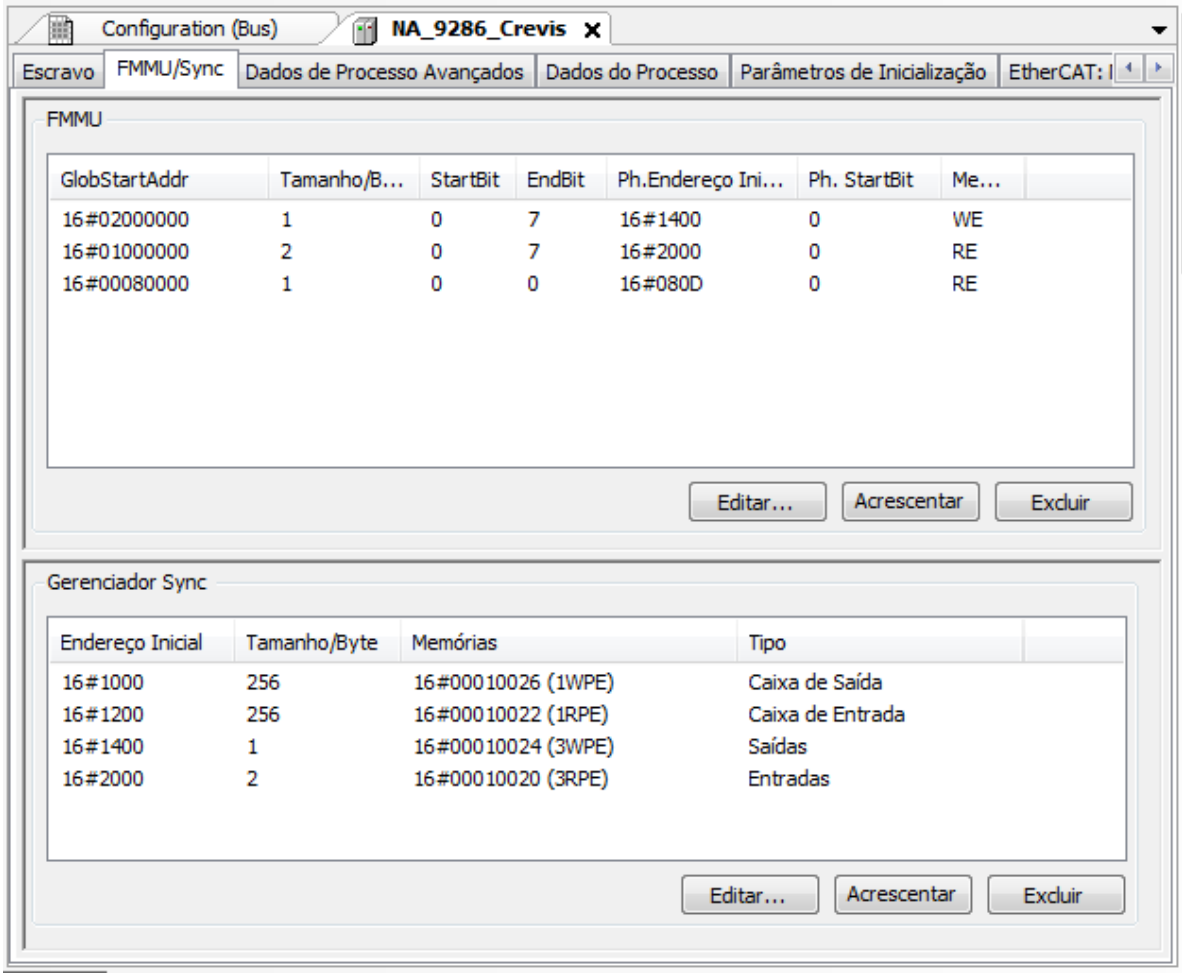

Figura 4-50. Diálogo FMMU/Sync

FMMU

Esta tabela mostra as Unidades do Gerenciador de Memória Fieldbus (FMMU - Fieldbus Memory Management Unit) do Escravo que são utilizadas para o tratamento dos dados do processo. Cada mapeamento do endereço lógico *GlobStartAddr* em um endereço físico *Ph. Endereço Inicial* é definido. É possível executar um mapeamento bit a bit. Novas unidades podem ser adicionadas e as já existentes podem ser editadas pelo diálogo Editar FMMU, que pode ser aberto através dos botões *Acrescentar* e *Editar...*

Gerenciador Sync

Esta tabela mostra o(s) Gerenciador(s) de Sincronização do Escravo. Para cada Tipo de Gerenciador disponível (Caixa de Entrada, Caixa de Saída, Entradas, Saídas) um Endereço Inicial físico, o tipo de Acesso, o Buffer e o Endereço Físico, onde as Interrupções tem de ser enviadas, são definidos. Novos Gerenciadores de Sincronização podem ser adicionados e os existentes podem ser abertos através dos botões *Acrescentar* e *Editar...*

Dados de Processo e Dados do Processo Avançados

A guia Dados do Processo do editor de configuração do Escravo EtherCAT mostra os dados do processo de entrada e de saída do Escravo, cada um definido por nome, tipo e índice do arquivo de descrição do dispositivo, como pode ser visto na Figura 4-51.

As entradas selecionadas (a serem lidas) e saídas (a serem escritas) do dispositivo estarão disponíveis no diálogo de Mapeamento de E/S como saídas e entradas para o CP, no qual as variáveis do projeto podem ser mapeadas.

A fim de modificar a seleção atual, você deve primeiro clicar com o mouse sobre a caixa de seleção dos dados selecionados no momento, para cancelar a seleção. Depois disso, você pode selecionar outros.

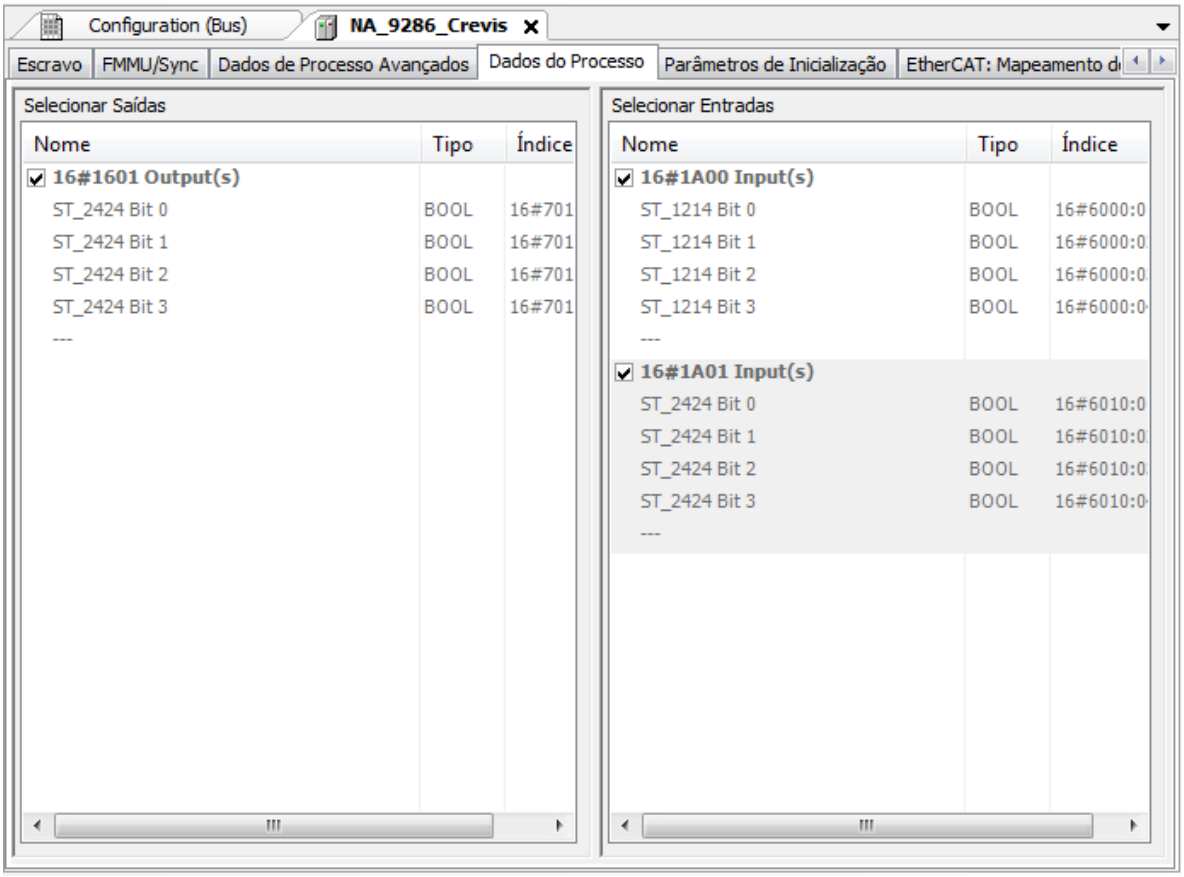

Figura 4-51. Diálogo Dados do Processo

O diálogo *Dados do Processo Avançados* só será visível no editor de configuração do Escravo EtherCAT se a opção *Habilitar Config. Avançadas* do Escravo estiver ativada. Ele fornece uma visão mais detalhada do dados do processo, além do que é apresentado na aba *Dados do Processo*. Além disso, nesta aba é possível habilitar o download da Atribuição PDO e a Configuração PDO.

ATENÇÃO: Se o Escravo não aceitar a configuração do PDO, ele ficará em modo pré-operacional e nenhuma troca de dados em tempo real será possível.

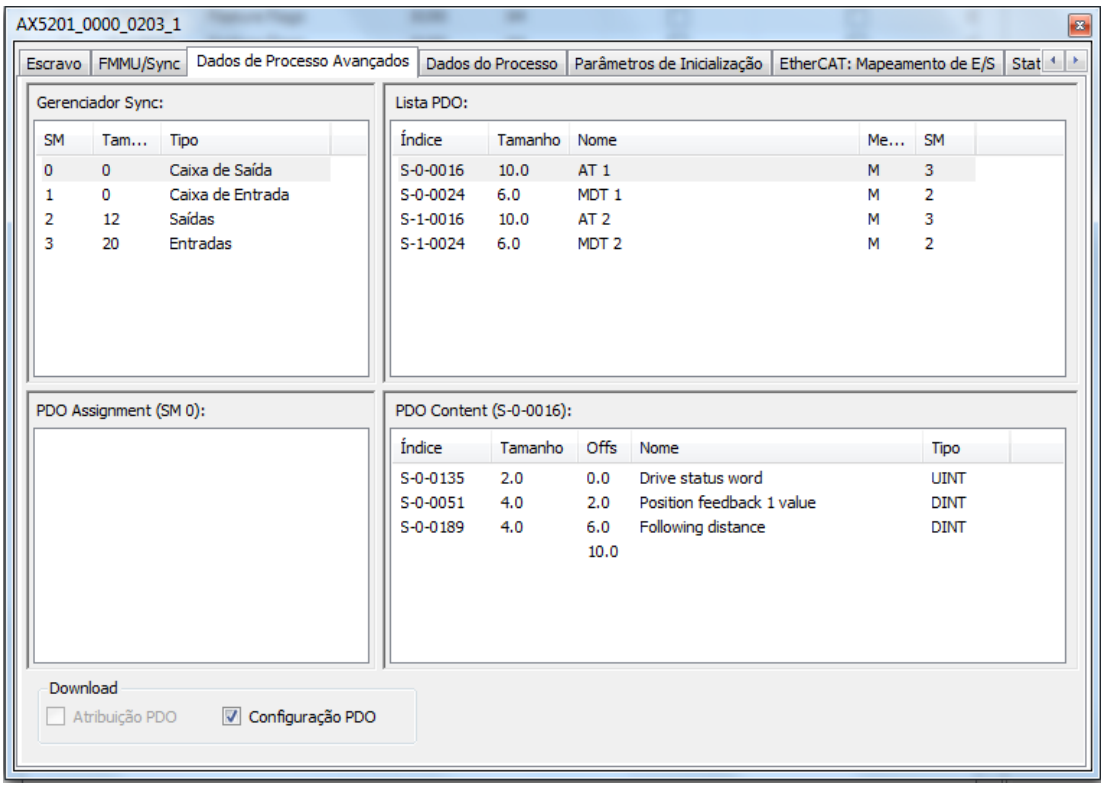

Figura 4-52. Diálogo Dados do Processo Avançados

Esta guia está dividida em quatro seções e duas opções:

- *Gerenciador Sync*: Lista de SM (Sync Manager) com o tamanho dos dados e tipo de PDO.
- *PDO Assignment*: Lista de PDO atribuídos ao *Gerenciador Sync* selecionado. A caixa de seleção ativa o PDO e canais E/S são criados. É parecido com as janelas de configuração de PDO simples. Aqui apenas PDOs podem ser ativados ou desativados.
- *Lista PDO*: Lista de todos os PDOs definidos no arquivo de descrição do dispositivo. PDO individuais podem ser apagados, editados ou adicionados por meio da execução do respectivo comando no menu de contexto.
- *PDO Content*: Exibe o conteúdo do PDO selecionado na seção acima. As inscrições podem ser apagadas, editadas ou adicionadas por meio da execução do respectivo comando a partir do menu de contexto.
- *PDO Assignment*: Se ativado um comando de escrita CoE será adicionado ao índice 0x1CXX para escrever a configuração PDO 0x16XX ou 0x1A00.
- *Configuração PDO*: Se ativado vários comandos da gravação CoE serão adicionados ao escrever o mapeamento PDO no Escravo.

ATENÇÃO: Se o Escravo não suportar a configuração PDO, o download desta pode resultar em um erro de Escravo. Esta função deve ser usada apenas por especialistas.

Editando a Lista PDO

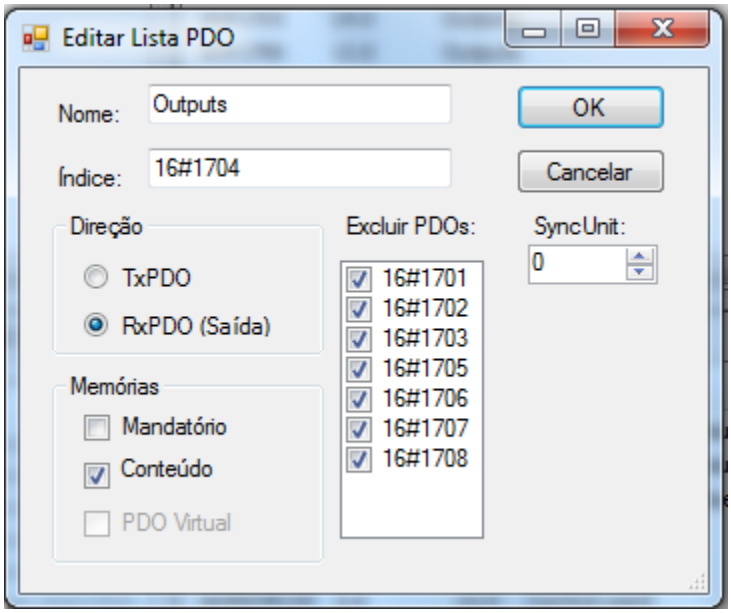

Figura 4-53. Diálogo Editar Lista PDO

Este diálogo é aberto através do menu de contexto da área *Lista PDO*, apresentada na Figura 4-52. Seguem algumas explicações das opções de configuração presentas neste diálogo.

- *Nome*: Nome da entrada PDO.
- *Índice*: Índice do PDO em edição.
- *TxPDO (Entrada)*: Se ativado, o PDO será transferido do Mestre para o Escravo.
- *RxPDO (Saída)*: Se ativado, o PDO será transferido do Escravo para o Mestre.
- *Mandatório*: O PDO é necessário e não pode ser desmarcado na área *PDO Assignment*.
- *Conteúdo*: O conteúdo do PDO é fixo e não pode ser alterado. Não é possível adicionar entradas no painel *PDO Content*.
- *PDO Virtual*: Reservado para uso futuro.
- *Excluir PDOs*: É possível definir PDOs que podem, ou não, serem selecionados juntos ao PDO em edição na área *PDO Assignment*, ou na guia *Dados do Processo*. Se um PDO é marcado nesta lista este não poderá ser selecionado, ficando cinza, no ambiente *PDO Assignment* quando o PDO em edição for selecionado.
- *SyncUnit*: ID do Gerenciador Sync do PDO para o qual deve ser atribuído.

Definição do Conteúdo do PDO

Este diálogo é acessado através do menu de contexto da área *PDO* Content e o seu conteúdo, além da possibilidade de acesso desta janela, varia de acordo com o Escravo EtherCAT utilizado.

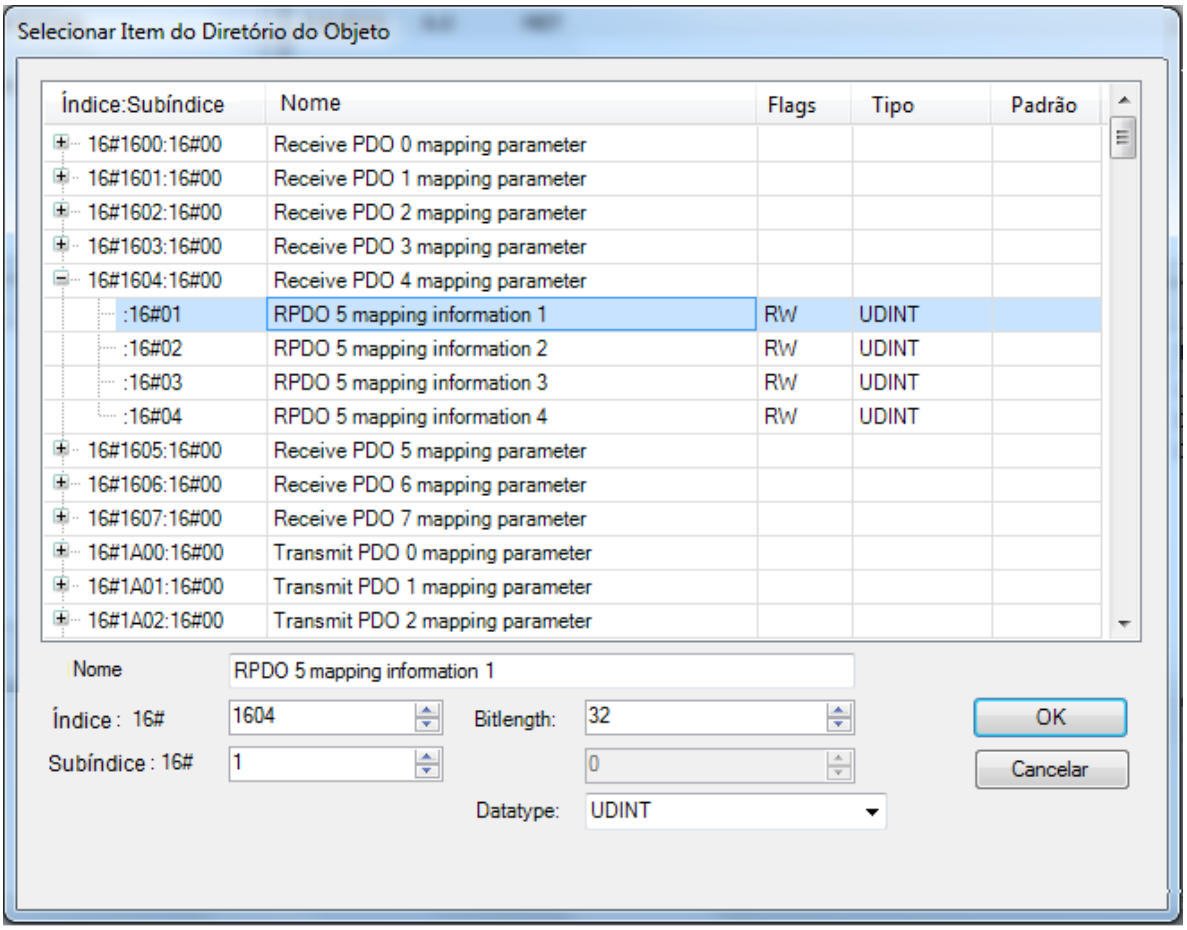

Figura 4-54. Diálogo Selecionar Item do Diretório do Objeto

Parâmetros de Inicialização

Na guia *Parâmetros de Inicialização* podem ser definidos parâmetros para o dispositivo, que serão repassados pelos *SDOs* (Service Data Objects) ou IDN na inicialização do sistema. As opções presentes neste ambiente, além da possibilidade de acesso, variam de acordo com o Escravo EtherCAT utilizado e estão presentes no seu *Arquivo de Descrição do Dispositivo*.

Online

O diálogo *Online* só será visível no editor de configuração Escravo EtherCAT, se a opção *Habilitar Config. Avançadas* do Escravo estiver ativada e a *Aplicação* estiver conectada ao dispositivo. Ele fornece uma visão com informações de status dos Escravos e funções para transferir arquivos para os Escravos sobre EtherCAT (FoE).

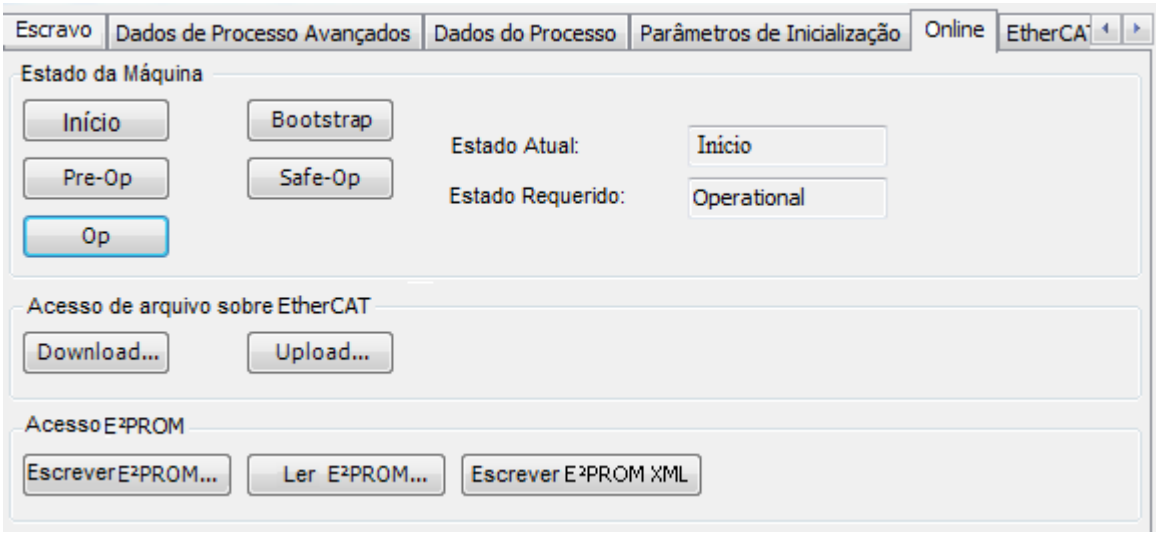

Figura 4-55. Diálogo Online

Esta guia é dividida nos seguintes grupos de funcionalidade:

- *Estado da Máquina*: Os botões *Início* (Init), *Pre-Op* (Pré- Operacional), *Op* (Operacional) e *Safe-Op* (Segurança Operacional) podem ser utilizados para o objetivo de debugging, eles fazem com que o Escravo vá para os respectivos estados.
- *Acesso através do arquivo EtherCAT*: Se você quiser transferir arquivos de firmware para o Escravo ou a partir dele, você tem que clicar no botão Bootstrap para mudar o Escravo para Modo Bootstrap. O Download e Upload de arquivos do firmware podem pode ser feitos com os botões correspondentes. Uma caixa de diálogo para salvar ou abrir o arquivo de firmware será aberta. Nesta caixa serão solicitados usuário e senha para executar a transferência de arquivos. Esta informação é fornecida pelo dispositivo Escravo e documentada no datasheet do mesmo.
- *Acesso E2PROM*: A configuração Escravo pode ser lida da E2PROM ou escrita na E2PROM. Aqui, tal qual a transferência de firmware, uma caixa de diálogo, para abrir ou salvar arquivos, será aberta. O comando escrever E2PROM XML pode ser usado para escrever a configuração do Escravo diretamente do arquivo XML para o dispositivo. Este comando só será habilitado se existirem dados de configuração no arquivo XML (seção <ConfigData>).

Mapeamento de E/S

Esta guia do editor de configuração do Escravo EtherCAT oferece a possibilidade de atribuir as variáveis de projeto para respectivas entradas e saídas do EtherCAT. Assim, as variáveis do Escravo EtherCAT podem ser controladas pela *Aplicação de Usuário*.

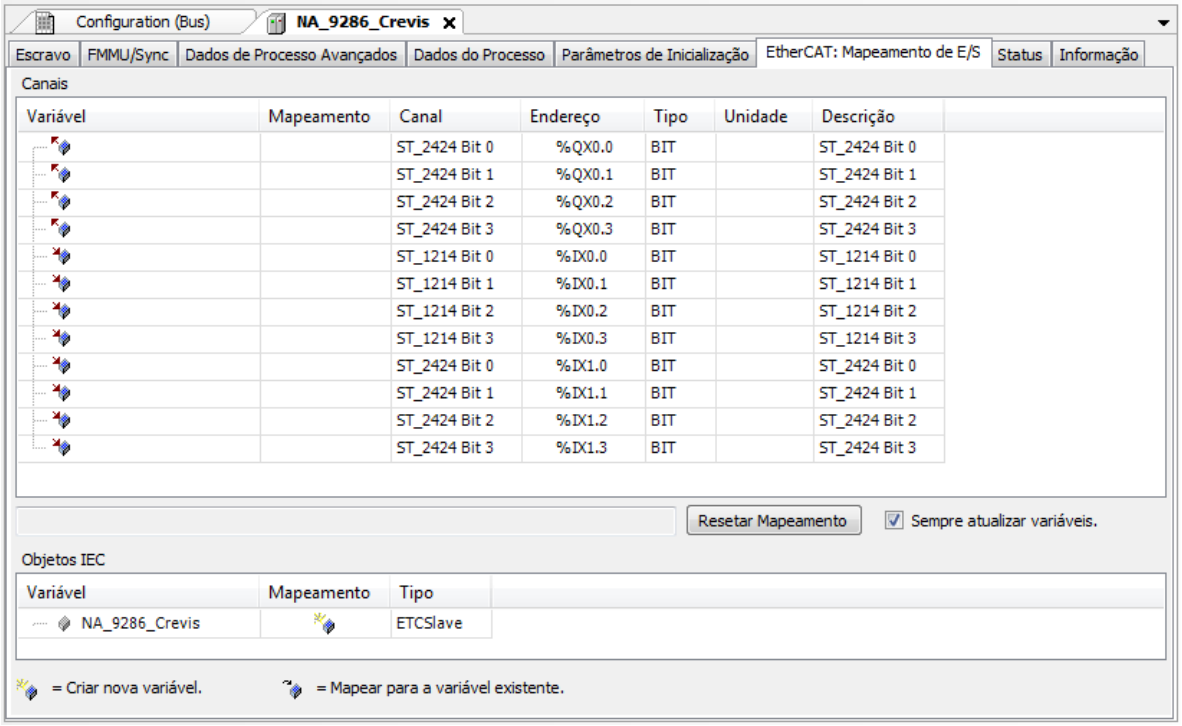

Figura 4-56. Diálogo Mapeamento de E/S

Guias Status e Informação

A guia *Status* do editor de configuração do Escravo EtherCAT fornece informações de estado (por exemplo, 'Executando', 'Parado') e mensagens de diagnóstico específicas do dispositivo e do sistema de barramento interno.

O diálogo *Informação*, presente no editor de configuração do Escravo EtherCAT, exibe, caso disponíveis, as seguintes informações gerais para o módulo: Nome, Fornecedor, Tipo, Número de Versão, Categorias, Número de Ordem, Descrição, Imagem.

Desempenho de Comunicação

Taxa de Comunicação de um Dispositivo MODBUS Servidor

Os dispositivos MODBUS configuráveis na UCP Nexto executam em segundo plano, com uma prioridade abaixo da aplicação de usuário e de forma cíclica. Sendo assim, seu desempenho irá variar de acordo com o tempo restante, levando em consideração a diferença entre o intervalo e tempo que a aplicação leva para ser executada. Por exemplo, um dispositivo MODBUS em uma aplicação que é executada a cada 100 ms, com um tempo de execução de 50 ms, irá ter um desempenho menor do que com uma aplicação de 50 ms executando a cada 200 ms de intervalo. Isto acontece porque, no segundo caso, a UCP terá um tempo maior entre cada ciclo da MainTask para executar as tarefas com prioridade mais baixa.

Também deve-se levar em conta o número de ciclos que o dispositivo, escravo ou servidor, leva para responder a uma requisição. Para processar e transmitir uma resposta, um MODBUS RTU Escravo irá levar dois ciclos (tempo do ciclo da tarefa MODBUS), enquanto que um MODBUS Ethernet Servidor levará apenas um ciclo. Porém esse é o tempo mínimo entre a recepção de uma requisição e o envio da resposta. Caso a requisição seja enviada logo após a execução de um ciclo da tarefa MODBUS, o tempo poderá ser equivalente a 2 ou 3 vezes o tempo de ciclo para o MODBUS Escravo, e de 1 a 2 vezes o tempo de ciclo para o MODBUS Servidor.

Neste caso: Tempo Máximo de Resposta = 3* (tempo de ciclo) + (tempo de execução das tarefas) + (tempo interframe chars) + (send delay).

Por exemplo, para uma tarefa MODBUS Ethernet Servidor com um ciclo de 50 ms, em uma aplicação executada por 60 ms a cada 100 ms, o servidor conseguirá executar apenas um ciclo entre cada ciclo da aplicação. Por outro lado, com a mesma aplicação, sendo executada por 60 ms, mas com um intervalo de 500 ms, o MODBUS terá um desempenho melhor, pois enquanto a aplicação não estiver em execução, o mesmo estará sendo executado a cada 50 ms e apenas a cada ciclo da MainTask ele irá demorar mais para ser executado. Para estes casos, o pior desempenho será a soma do tempo de execução da aplicação do usuário com o tempo do ciclo da tarefa MODBUS.

Para os dispositivos mestre e cliente o princípio de funcionamento é exatamente igual, mas levando em consideração o tempo de polling da relação MODBUS e não o tempo do ciclo da tarefa MODBUS. Para estes casos, o pior desempenho de uma relação será ser executada após o seu tempo de polling, somado ao tempo de execução da aplicação de usuário.

É importante ressaltar que o número de dispositivos MODBUS em execução também irá alterar o seu desempenho. Em uma aplicação de usuário com tempo de execução de 60 ms e intervalo de 100 ms, irá restar 40 ms para a UCP executar todas as tarefas de menor prioridade. Portanto, uma UCP com apenas um MODBUS Ethernet Servidor terá um desempenho maior do que uma UCP que utilize quatro destes dispositivos.

Interfaces Locais da UCP

Para um dispositivo MODBUS Ethernet Servidor, podemos afirmar que ele é capaz de responder a um número x de requisições por segundo, ou seja, será capaz de transferir n bytes por segundo, dependendo do tamanho de cada requisição. Quanto menor for o ciclo da tarefa do Servidor MODBUS, maior será o impacto do número de conexões em sua taxa de resposta. Porém, para tempos de ciclo menores que 20 ms este impacto não é linear, devendo ser consultada a Tabela 4-75 para informações.

A Tabela 4-75 exemplifica o número de requisições respondidas por um Servidor MODBUS inserido em uma interface local da UCP, em função do tempo de ciclo configurado para a tarefa e do número de conexões ativas efetuando requisições:

Número de Conexões	Requisições respondidas por segundo com tempo de ciclo da tarefa MODBUS em 5 ms	Requisições respondidas por segundo com tempo de ciclo da tarefa MODBUS em 20 ms
1 Conexão	160	47
2 Conexões	300	95
3 Conexões	395	142
4 Conexões	470	190
5 Conexões	515	237
6 Conexões	570	285
7 Conexões	605	332
8 Conexões	640	380
9 Conexões	665	427
10 Conexões	690	475

Tabela 4-75. Taxa de Comunicação de um Servidor MODBUS em uma Interface Local

ATENÇÃO: Os desempenhos de comunicação mencionados neste capítulo são exemplos utilizando uma UCP apenas com um dispositivo MODBUS TCP Servidor, sem nenhuma lógica na aplicação que possa atrasar a comunicação. Portanto, estes desempenhos devem ser tomados como máximos.

Já para tempos de ciclo iguais ou maiores do que 20 ms, o crescimento da taxa de respostas é linear, podendo ser calculada através da fórmula:

N =- C x (Z – (Z / (T x 1000)))

Z = 1 / T

Onde *N* é o número médio de respostas por segundo, *C* é o número de conexões e T é igual ao tempo de ciclo da tarefa MODBUS (em segundos).

Tomando como exemplo um Servidor MODBUS com uma conexão e um tempo de ciclo de 50 ms temos:

C = 1; **T** = 0,05 s; **Z** = 1 / 0,05 = 20;

N = 1 x (20 – (20 / (0,05 x 1000))) = 1 x (20 – (20 / 50)) = 1 x (20 – 0,4) = 1 x 19,6

N = 19,6

Ou seja, nesta configuração o Servidor MODBUS será capaz de responder, em média, 19 requisições por segundo.

Caso este valor obtido seja multiplicado pelo número de bytes em cada requisição, será obtida uma taxa de transferência de n bytes por segundo.

Interfaces Remotas

O desempenho de um dispositivo MODBUS Servidor em uma interface Ethernet remota é parecido com o desempenho nas interfaces locais da UCP.

Porém, devido ao tempo de comunicação entre a UCP e os módulos, o máximo desempenho será limitado. Para apenas uma conexão aberta, o número de respostas é limitado em, no máximo, 18 respostas por segundo. Com mais conexões abertas, o número de respostas aumentará linearmente, exatamente conforme para as interfaces locais, porém sendo limitado a, no máximo, 90 respostas por segundo. Portanto, para uma interface Ethernet remota teremos as seguintes formas para calcular seu desempenho:

Para $T \leq 55\ ms$ usa-se:

N = C x (18,18 – (18,18 / (0,055 x 1000)))

E para $T \geq 55\ ms$ usa-se:

N = C x (Z – (Z / (T x 1000)))

Onde *N* é o número médio de respostas por segundo, *C* é o número de conexões e T é igual ao tempo de ciclo da tarefa MODBUS (em segundos).

Deve-se prestar atenção ao fato de que o desempenho máximo de um dispositivo MODBUS Servidor em uma interface Ethernet remota será de 90 respostas de requisições por segundo.

Taxa de Comunicação de um Dispositivo com Servidor OPC

Para os testes de desempenho da comunicação com Servidor OPC foram criados projetos para o CP declarando variáveis do tipo INT e separando-as em POUs com 1.000 variáveis cada. Em todos os cenários foram utilizados projetos com profile Single e o Intervalo da MainTask configurado em 100 ms. Para realizar a comunicação, foi utilizado o atributo {attribute 'symbol' := 'readwrite'}, afim de disponibilizar os dados no Servidor OPC. Também é importante ressaltar que as medições foram feitas com o software MasterTool desconectado da CPU e com a duração da MainTask ajustada para consumir 5%, 50% e 80% do Intervalo configurado, como apresentado na Tabela 4-76.

No ponto de vista do OPC Cliente, foi utilizado um driver num sistema SCADA. O tempo atualização configurado foi de 50 ms. Os resultados do desempenho nestas condições estão descritos na Tabela 4-76.

Número Total de variáveis no projeto do CP	Tempo de atualização das variáveis no OPC Cliente		
	5% da CPU Ocupada	50% da CPU Ocupada	80% da CPU Ocupada
1.000	600 ms	800 ms	1400 ms
2.000	800 ms	900 ms	2800 ms
5.000	1000 ms	2000 ms	6500 ms
10.000	2000 ms	4000 ms	13700 ms

15.000	3200 ms	6400 ms	20000 ms
20.000	4000 ms	8100 ms	25000 ms

Tabela 4-76. Taxa de Comunicação de um Servidor OPC

Disparo de Relações MODBUS Cliente de Forma Acíclica

Para disparar relações MODBUS Cliente de forma acíclica, sugere-se o seguinte método, que pode ser implementado de maneira simples no programa aplicativo do usuário:

- Definir tempo máximo de polling para as relações;
- Manter a relação normalmente desabilitada;
- Habilitar a relação no momento em que se deseja executá-la;
- Esperar pela confirmação de término da execução da relação, e neste momento desabilitá-la novamente.

Desempenho do Sistema

Em casos onde a aplicação possui apenas uma tarefa de usuário MainTask, responsável pela execução de uma única unidade de programação do tipo Programa denominada MainPrg (Como no perfil Simples), o CP consome um determinado tempo para que a tarefa seja processada. A esse tempo damos o nome de Tempo de Execução.

Em uma aplicação, podemos conhecer o Tempo de Execução médio da aplicação usando o MasterTool IEC XE, na Árvore de Dispositivos, no item Device, no seguinte caminho:

PLC Logic-> Application-> Task Configuration na aba Monitor, na coluna Tempo de Ciclo Médio.

Deve-se atentar ao Tempo de Execução para que ele não ultrapasse 80% do intervalo configurado na tarefa de usuário MainTask. Por exemplo, em uma aplicação onde o intervalo é de 100 ms, um Tempo de Execução adequado é de até 80 ms. Isso se deve ao fato de que a UCP necessita de um tempo para a execução de outras tarefas como o processamento da comunicação, tratamento do visor e cartão de memória, e essas tarefas também acontecem dentro do intervalo (os 20% restantes do Tempo de Execução).

ATENÇÃO:
Para tempos de ciclo muito altos (tipicamente tempos maiores que 300 ms), mesmo que o percentual de 80% seja respeitado, pode haver uma diminuição no tempo de resposta do visor e do botão de diagnóstico. Caso o percentual de 80% não seja respeitado e o tempo de execução da tarefa de usuário se aproxime ou exceda o intervalo configurado da MainTask, o visor e o botão de diagnóstico podem não responder, uma vez que sua prioridade na execução do sistema é menor do que as tarefas de usuário. Caso uma aplicação com erros seja carregada na UCP, pode ser necessário reinicializá-la sem carregar esta aplicação, como descrito na seção Log de Sistema.

ATENÇÃO:
Os logs de sistema das UCPs da série Nexto, a partir da versão de firmware 1.4.0.33 são recarregados no caso de uma reinicialização da UCP ou por uma reinicialização do Runtime System, isto é, será possível visualizar os logs mais antigos quando uma dessas condições ocorrer.

Escaneamento de E/S

Para um projeto que utilize de módulos de E/S digitais, estando estes inseridos no barramento e declarados no projeto, o tempo da MainTask irá aumentar de acordo com o número de módulos. A Tabela 4-77 exemplifica o tempo médio que é acrescido à MainTask:

Módulos declarados no barramento	Acréscimo de tempo no ciclo da MainTask (µs)
5	300
10	700
20	1000

Tabela 4-77. Tempo de Escaneamento de E/S

Em projetos que utilizem E/S remotas, como, por exemplo, utilizando o módulo Mestre Profibus DP NX5001, o manual do respectivo módulo deve ser consultado para informações sobre desempenho e influências do módulo na execução das tarefas de usuário.

Cartão de Memória

A transferência de dados envolvendo o cartão de memória é realizada em segundo plano pela UCP, pois esta dá prioridade para a execução da aplicação de usuário e processamento da comunicação. Desta forma, a transferência de arquivos para o cartão poderá sofrer um acréscimo de tempo significativo, dependendo do tempo de execução da aplicação do usuário.
O tempo necessário para ler/escrever arquivos no cartão será afetado diretamente pelo tempo de execução da aplicação de usuário, uma vez que essa aplicação tem prioridade na execução.

Para mais informações sobre a utilização do cartão de memória, ver capítulo Configuração - Cartão de Memória.

Relógio RTC

As UCPs da Série Nexto possuem um relógio interno que pode ser utilizado através da biblioteca NextoStandard.lib. Essa biblioteca é carregada automaticamente durante a criação de um novo projeto (para realizar o procedimento de inserção de uma biblioteca, consultar capítulo Programação Inicial – Bibliotecas). A Figura 4-57 mostra os blocos e funções disponibilizados:

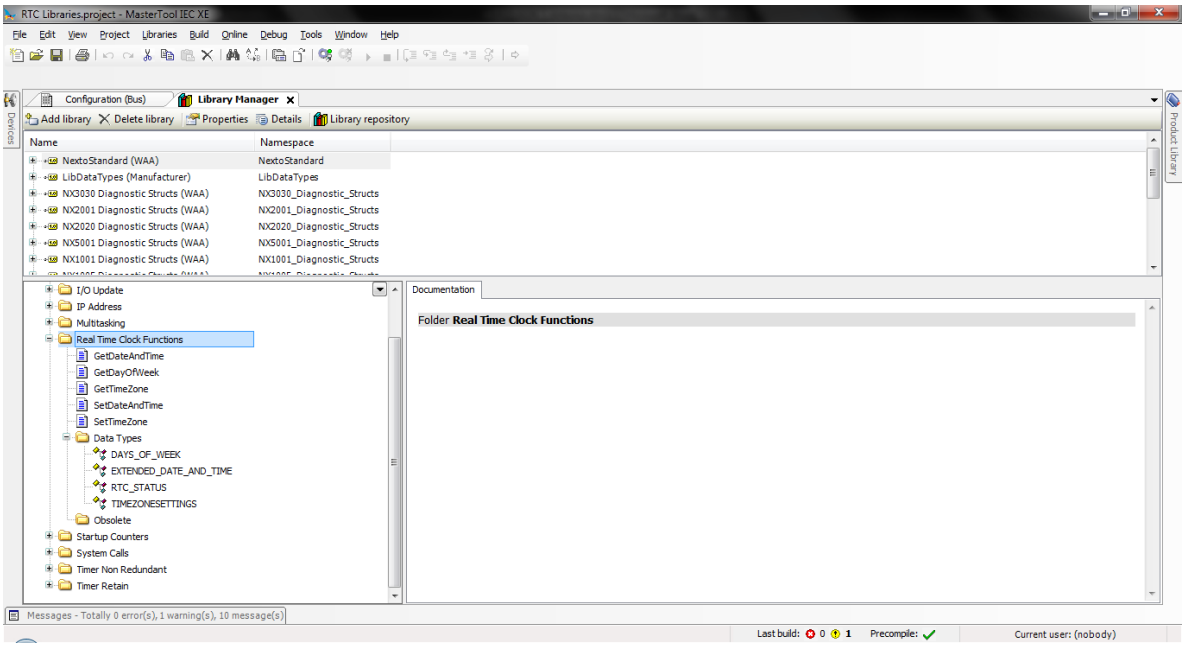

Figura 4-57. Blocos e Funções de Escrita e Leitura do Relógio

ATENÇÃO:
Blocos funcionais para Leitura e Escrita do RTC, disponíveis em versões anteriores a 2.00 do MasterTool IEC XE tornaram-se obsoletos da versão 2.00 em diante, os blocos que ficaram obsoletos são:
NextoGetDateAndTime, NextoGetDateAndTimeMs e NextoGetTimeZone, NextoSetDateAndTime, NextoSetDateAndTimeMs e NextoSetTimeZone.

Blocos Funcionais e Funções para Leitura e Escrita do RTC

Entre outros blocos funcionais e funções, existem alguns muito importantes utilizados para a leitura do relógio (GetDateAndTime, GetDayOfWeek e GetTimeZone) e para configurar novos valores de data e hora (SetDateAndTime e SetTimeZone). Estas funções utilizam sempre o horário local, isto é, levam em consideração o valor definido para o TimeZone.

Abaixo, descrevem-se os procedimentos utilizados para configurar as funções.

ATENÇÃO:
Os blocos funcionais obsoletos para leitura e escrita do RTC (NextoGetDateAndTime, NextoGetDateAndTimeMs, NextoSetDateAndTime e NextoSetDateAndTimeMs) não podem ser utilizados na área de dados redundantes em projetos redundantes. Os mesmos devem ser utilizados em POUs não redundantes, como a POU NonSkippedPrg. Para mais detalhes sobre o funcionamento da POU NonSkippedPrg pode ser encontrado em Programa NonSkippedPrg.

Funções de Leitura do RTC

A leitura do relógio pode ser feita através das funções a seguir:

GetDateAndTime

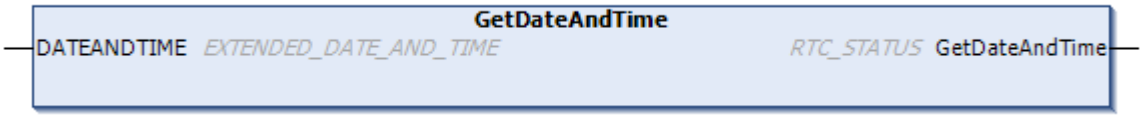

Figura 4-58. Leitura da Data e Hora

Parâmetros de entrada e saída	Tipo	Descrição
DATEANDTIME	EXTENDED_DATE_AND_TIME	Esta variável retorna o valor de data e hora do RTC no formato apresentado na Tabela 4-87.

Tabela 4-78. Parâmetros de Entrada e Saída GetDateAndTime

Parâmetros de saída	Tipo	Descrição
GETDATEANDTIME	RTC_STATUS	Retorna o estado de erro da função, ver Tabela 4-89.

Tabela 4-79. Parâmetros de Saída GetDateAndTime

Exemplo de utilização em Linguagem ST:

```
PROGRAM MainPrg
VAR
    Result : RTC_STATUS;
    DATEANDTIME : EXTENDED_DATE_AND_TIME;
    xEnable : BOOL;
```

```
END_VAR
-------------------------------------------------------------------------
IF xEnable = TRUE THEN

   Result := GetDateAndTime(DATEANDTIME);
   xEnable := FALSE;
END_IF
```

GetTimeZone

A função a seguir faz a leitura das configurações de fuso horário, esta função está diretamente relacionada com o tempo de fuso horário configurado no serviço de sincronismo do SNTP:

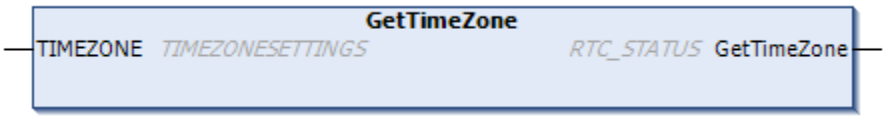

Figura 4-59. Leitura das Configurações de Fuso Horário

Parâmetros de entrada e saída	Tipo	Descrição
TimeZone	TIMEZONESETTINGS	Essa variável apresenta a leitura das configurações de fuso horário.

Tabela 4-80. Parâmetros de Entrada e Saída GetTimeZone

Parâmetros de saída	Tipo	Descrição
GetTimeZone	RTC_STATUS	Retorna o estado de erro da função, ver Tabela 4-89.

Tabela 4-81. Parâmetros de Saída GetTimeZone

Exemplo de utilização em Linguagem ST:

```
PROGRAM MainPrg
VAR

   GetTimeZone_Status      : RTC_STATUS;
   TimeZone                : TIMEZONESETTINGS;
   xEnable : BOOL;
END_VAR
-------------------------------------------------------------------------
IF xEnable = TRUE THEN

   GetTimeZone_Status := GetTimeZone(TimeZone);
   xEnable := FALSE;
END_IF
```

GetDayOfWeek

A função GetDayOfWeek é utilizada para fazer a leitura do dia da semana.

Figura 4-60. Leitura do Dia da Semana

Parâmetros de saída	Tipo	Descrição
GetDayOfWeek	DAYS_OF_WEEK	Retorna o dia da semana. Ver seção Estruturas de Dados do RTC - DAYS_OF_WEEK

Tabela 4-82. Parâmetros de Saída GetDayOfWeek

Quando chamada, a função lerá o dia da semana e preencherá a estrutura DAYS_OF_WEEK.

Exemplo de utilização em Linguagem ST:

```
PROGRAM MainPrg
VAR
   DayOfWeek   : DAYS_OF_WEEK;
END_VAR
------------------------------------------------------------------------------
DayOfWeek := GetDayOfWeek();
```

Blocos Funcionais e Funções de Escrita e Configuração do RTC

As configurações de relógio são feitas através das funções e blocos funcionais a seguir:

SetDateAndTime

O Bloco Funcional SetDateAndTime é utilizado para realizar a configuração do relógio:

Figura 4-61. Configuração de Data e Hora com Milissegundos

Parâmetros de entrada	Tipo	Descrição
REQUEST	BOOL	Essa variável, quando recebe uma borda de subida, habilita a escrita do relógio.
DATEANDTIME	EXTENDED_DATE_AND_TIME	Recebe os valores de data e hora com milissegundos. Ver seção Estruturas de Dados do RTC - EXTENDED_DATE_AND_TIME.

Tabela 4-83. Parâmetros de Entrada SetDateAndTime

Parâmetros de saída	Tipo	Descrição
DONE	BOOL	Essa variável, quando verdadeira, indica que a ação foi completada com sucesso.
EXEC	BOOL	Essa variável, quando verdadeira, indica que a função está processando os valores.
ERROR	BOOL	Essa variável, quando verdadeira, indica que ocorreu algum erro durante a Escrita.
STATUS	RTC_STATUS	Retorna o erro ocorrido durante a configuração. Ver seção Estruturas de Dados do RTC - RTC_STATUS.

Tabela 4-84. Parâmetros de Saída SetDateAndTime

Quando ocorrer uma borda de subida na entrada REQUEST, o bloco funcional irá escrever o novo valor DATEANDTIME no relógio. Caso a escrita seja realizada com sucesso, a saída DONE será igual a TRUE. Caso contrário, a saída ERROR será igual a TRUE e o erro será apresentado na variável STATUS.

Exemplo de utilização em Linguagem ST:

```
PROGRAM MainPrg
VAR
    SetDateAndTime : SetDateAndTime;
    xRequest : BOOL;
    DateAndTime : EXTENDED_DATE_AND_TIME;
    xDone : BOOL;
    xExec : BOOL;
    xError : BOOL;
    Status : RTC_STATUS;
    xWrite : BOOL;
END_VAR
-------------------------------------------------------------------------
IF (xWrite = TRUE) THEN
    SetDateAndTime(
        request := xRequest,
        done => xDone,
        exec => xExec,
        error => xError,
        status => status,
        DateAndTime := DateAndTime);
    xWrite := FALSE;
END_IF
```

ATENÇÃO: Se o usuário tentar escrever valores de hora fora do intervalo do RTC, os valores serão convertidos para valores válidos, desde que não ultrapasse a faixa válida de 01/01/2000 até 31/12/2035. Por exemplo, se o usuário tentar escrever o valor 2000 ms, o mesmo será convertido para 2 segundos, se escrever o valor 100 segundos, o mesmo será convertido para 1 min e 40 segundos. Se escrever o valor de 30 horas, o mesmo será convertido para 1 dia e 6 horas, e assim por diante.

SetTimeZone

A função SetTimeZone realiza a escrita das configurações de fuso horário:

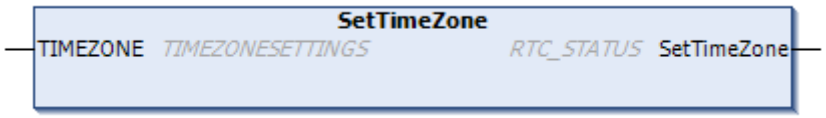

Figura 4-62. Configuração de Fuso Horário

Parâmetros de entrada	Tipo	Descrição
TIMEZONE	TIMEZONESETTINGS	Estrutura com o valor de fuso horário a ser configurado. Ver seção Estruturas de Dados do RTC - TIMEZONESETTINGS.

Tabela 4-85. Parâmetros de Entrada SetTimeZone

Parâmetros de saída	Tipo	Descrição
SetTimeZone	RTC_STATUS	Retorna o erro ocorrido durante a leitura/configuração. Ver seção Estruturas de Dados do RTC - RTC_STATUS.

Tabela 4-86. Parâmetros de Saída SetTimeZone

Quando chamada, a função, irá configurar o valor de TIMEZONE como a nova configuração de fuso horário do sistema. O resultado da configuração é retornado pela função.

Exemplo de utilização em Linguagem ST:

```
PROGRAM MainPrg
VAR
   Status : RTC_STATUS;
   TimeZone : TIMEZONESETTINGS;
   xWrite : BOOL;
END_VAR
-------------------------------------------------------------------------
//FB SetTimeZone
IF (xWrite = TRUE) THEN
   Status := SetTimeZone(TimeZone);
   xWrite := FALSE;
END_IF
```

ATENÇÃO: Para realizar o acerto do relógio, devem-se utilizar valores de hora e datas dentro da seguinte faixa válida: 00:00:00 horas de 01/01/2000 até 23:59:59 horas de 31/12/2035, caso contrário, será reportado um erro através do parâmetro de saída STATUS. Para maiores detalhes do parâmetro de saída STATUS, consultar a seção RTC_STATUS.

Estruturas de Dados do RTC

Os blocos funcionais de leitura e configuração do RTC das UCPs da série Nexto utilizam as seguintes estruturas de dados na sua configuração:

EXTENDED_DATE_AND_TIME

Esta estrutura é utilizada para armazenar a data do RTC quando utilizados os blocos funcionais para leitura/configuração da data com precisão de milissegundos e é descrita na Tabela 4-87:

Estrutura	Tipo	Variável	Descrição
EXTENDED_DATE_AND_TIME	BYTE	byDayOfMonth	Armazena o dia do mês da data configurada.
	BYTE	ByMonth	Armazena o mês da data configurada.
	WORD	wYear	Armazena o ano da data configurada.
	BYTE	byHours	Armazena a hora da data configurada.
	BYTE	byMinutes	Armazena os minutos da data configurada.
	BYTE	bySeconds	Armazena os segundos da data configurada.
	WORD	wMilliseconds	Armazena os milissegundos da data configurada.

Tabela 4-87. Estrutura EXTENDED_DATE_AND_TIME

DAYS_OF_WEEK

Esta estrutura é utilizada para armazenar o dia da semana quando utilizada a função para leitura do dia da semana:

Enumerador	Valor	Descrição
DAYS_OF_WEEK	0	INVALID_DAY
	1	SUNDAY
	2	MONDAY
	3	TUESDAY
	4	WEDNESDAY
	5	THURSDAY

	6	FRIDAY
	7	SATURDAY

Tabela 4-88. Estrutura DAYS_OF_WEEK

RTC_STATUS

Este enumerador é utilizado para retornar o tipo de erro na configuração ou leitura do RTC e é descrito na Tabela 4-89:

Enumerador	Valor	Descrição
RTC_STATUS	NO_ERROR (0)	Não há erro.
	UNKNOWN_COMMAND (1)	Comando desconhecido.
	DEVICE_BUSY (2)	Dispositivo está ocupado.
	DEVICE_ERROR (3)	Dispositivo com erro.
	ERROR_READING_OSF (4)	Erro na leitura do sinalizador de data e hora válidos.
	ERROR_READING_RTC (5)	Erro na leitura da data e hora.
	ERROR_WRITING_RTC (6)	Erro na escrita da data e hora.
	ERROR_UPDATING_ SYSTEM_TIME (7)	Erro na atualização de data e hora do sistema.
	INTERNAL_ERROR (8)	Erro interno.
	INVALID_TIME (9)	Data e hora inválidas.
	INPUT_OUT_OF_RANGE (10)	Fora do limite de Data e hora válidas para o sistema.
	SNTP_NOT_ENABLE (11)	Erro gerado quando o serviço SNTP não está habilitado e é feita uma tentativa de ler ou modificar o fuso horário.

Tabela 4-89. RTC_STATUS

TIMEZONESETTINGS

Esta estrutura é utilizada para armazenar o valor do fuso horário nas requisições de leitura/configuração dos blocos funcionais do RTC e é descrita na Tabela 4-6:

Estrutura	Tipo	Variável	Descrição
TIMEZONE SETTINGS	INT	iHour	Hora do fuso horário configurado.
	INT	iMinutes	Minuto do fuso horário configurado.

Tabela 4-90. TIMEZONESETTINGS

Nota:

Blocos funcionais de escrita e leitura de data e hora: Bibliotecas diferentes da NextoStandard, que tenham blocos funcionais ou funções que possam fazer acesso de leitura e escrita da data e hora no sistema, não são indicados. A biblioteca NextoStandard possui as interfaces adequadas para escrever e ler a data e hora do sistema adequadamente e informar os diagnósticos corretos.

Memória de Arquivos de Usuário

As UCPs da Série Nexto possuem uma área de memória destinada ao armazenamento de dados de uso geral, ou seja, o usuário poderá gravar diversos arquivos de qualquer formato na memória da UCP. Esta área de memória pode variar de acordo com o modelo de UCP utilizado (consultar capítulo Descrição Técnica – Características Gerais – Características Específicas).

Para usar esta área, o usuário deverá acessar um projeto no software MasterTool IEC XE e clicar na Árvore de Dispositivos, localizada à esquerda do programa. Deverá dar dois cliques sobre o item Device e, após selecionar a UCP na aba Configurações de Comunicação que será aberta, selecionar a aba Arquivos e clicar em Atualizar, tanto na coluna de arquivos do computador (esquerda), como na coluna de arquivos da UCP selecionada (direita), conforme mostram as indicações da Figura 4-63.

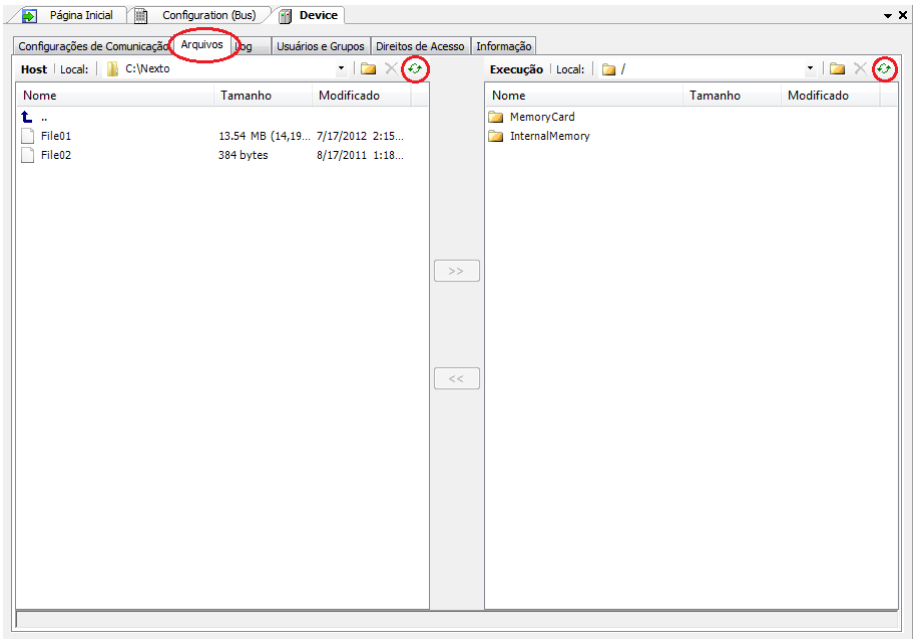

Figura 4-63. Acesso aos Arquivos de Usuário

Após atualizar a coluna de arquivos da UCP, será exibido o diretório raiz de arquivos armazenados na UCP e poderá ser selecionada a pasta para onde os arquivos serão transferidos. A pasta "InternalMemory" é uma pasta padrão, a ser utilizada para armazenar arquivos na memória interna da UCP, uma vez que não é possível transferir arquivos para o diretório raiz. Caso seja necessário, podem ser criadas outras pastas no diretório raiz ou subpastas dentro da pasta "InternalMemory". Já a pasta "MemoryCard" é o diretório onde o cartão de memória estará montado, caso o mesmo esteja inserido na UCP. Arquivos transferidos para a pasta "MemoryCard" estarão sendo transferidos diretamente para dentro do cartão de memória.

ATENÇÃO: No caso em que o cartão de memória seja inserido após a inicialização da UCP, será requisitado um usuário e senha para realizar o acesso e/ou transferências de arquivos do MasterTool IEC XE para o cartão de memória ou vice-versa. O usuário padrão com privilégios para acesso à UCP é "Owner" e a senha padrão desse usuário é "Owner".

Para realizar a transferência de algum arquivo do microcomputador para a UCP, basta selecionar o arquivo desejado na coluna da esquerda e pressionar o botão ">>", localizado no centro da tela, conforme Figura 4-64. O tempo de transferência irá variar de acordo com o tamanho do arquivo e com o tempo de ciclo (execução) da aplicação atual da UCP, podendo levar vários minutos.

O usuário não precisa estar em Modo Run ou conectado à UCP para realizar as transferências, pois ela possui a capacidade de se conectar automaticamente quando o usuário realizar a transferência.

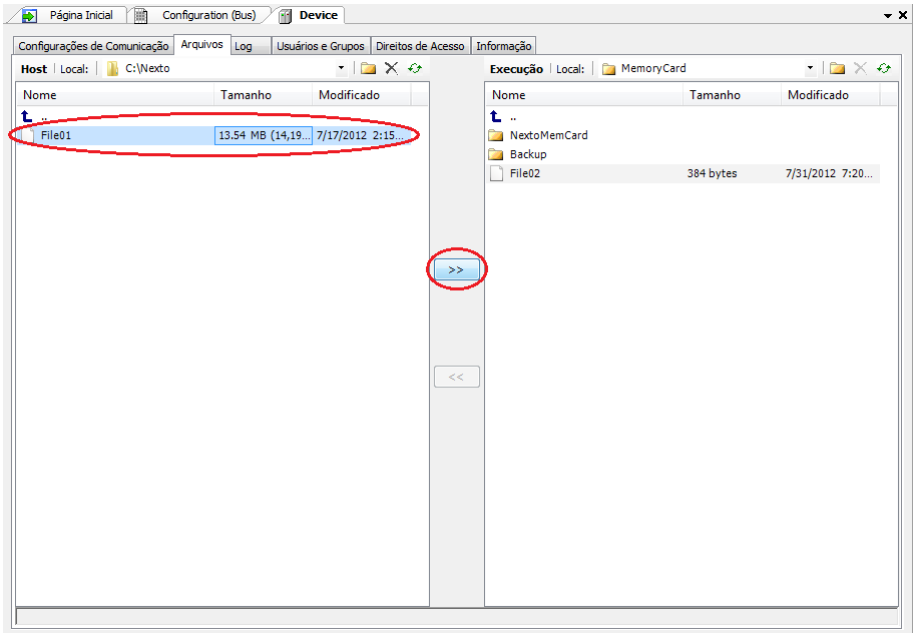

Figura 4-64. Transferindo Arquivos

ATENÇÃO: Os arquivos contidos dentro da pasta de um projeto criado pela ferramenta MasterTool IEC XE possuem nomes especiais reservados pelo sistema, desta forma não podem ser transferidos através da aba Arquivos. Caso o usuário deseje transferir um projeto para a memória de usuário, será necessário compactar a pasta e então transferir o arquivo compactado (*.zip por exemplo).

Caso seja necessário transferir documentos da UCP para o microcomputador em que está instalado o software MasterTool IEC XE o usuário deve realizar um procedimento muito semelhante ao anterior, ou seja, selecionando o arquivo na coluna da direita e pressionar o botão "<<", localizado no centro da tela.

Além disso, o usuário possui algumas opções de operação da área de armazenamento de arquivos, são elas:

- Novo diretório : permite a criação de uma nova pasta na área de memória de usuário.
- Excluir item : permite a exclusão de arquivos nos diretórios da área de memória de usuário.
- Atualizar : permite atualizar, na tela do MasterTool IEC XE, os arquivos presentes na memória de usuário e no microcomputador.

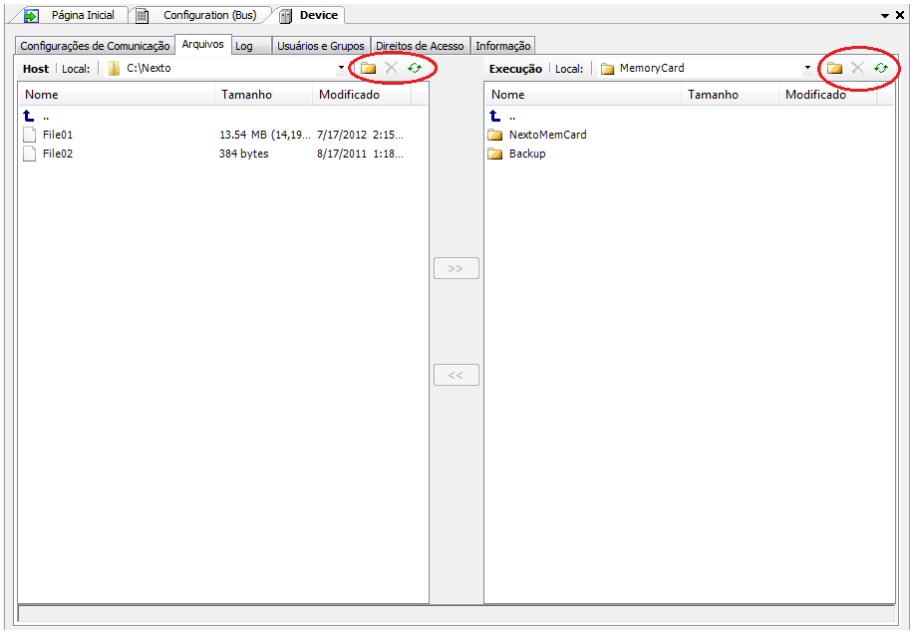

Figura 4-65. Opções de Utilização

ATENÇÃO: Para uma UCP em Modo Stop ou sem nenhuma aplicação, a taxa de transferência para a memória interna é de aproximadamente 150 Kbytes/s.

Cartão de Memória

Entre outras memórias, os modelos NX3010, NX3020 e NX3030 das UCPs da Série Nexto possibilitam ao usuário a utilização de um cartão de memória, definido conforme as características do capítulo Descrição Técnica - Interface do Cartão de Memória, o qual serve para armazenar, entre outros arquivos, o projeto e a aplicação que está na memória interna da UCP.

Quando o cartão for inserido na UCP e estiver com sistema de arquivos diferente de FAT32, ela automaticamente identifica e pergunta ao usuário se ele deseja formatar. Em caso negativo, ele não poderá utilizar o cartão (o cartão não será montado, aparecerá uma mensagem dizendo que o formato não foi reconhecido, e o visor não indicará a presença do cartão). Caso seja selecionada a opção de formatação, a UCP irá levar alguns minutos, dependendo do tempo de ciclo (execução) da aplicação rodando na UCP, para executar a operação. Assim que o cartão de memória for montado, a UCP irá ler informações gerais do mesmo, deixando o acesso ao cartão de memória mais lento nos primeiros minutos. Esse procedimento é feito apenas quando o cartão é inserido ou a UCP reinicializada.

ATENÇÃO: Recomenda-se realizar o procedimento de formatação do cartão de memória diretamente na UCP Nexto para evitar possíveis problemas de utilização, aumento do tempo de montagem ou até mesmo funcionamento incorreto. Não é recomendado remover o cartão de memória ou desenergizar a UCP durante a formatação ou durante a transferência de arquivos, pois pode causar a perda de dados bem como danos irreversíveis ao cartão.

Durante a configuração do projeto, dentro do software MasterTool IEC XE, o usuário habilita a opção de cópia do projeto da UCP para o cartão de memória ou a cópia do projeto que está no cartão para a UCP. Nessa mesma tela o usuário poderá configurar senhas de acesso, controlando o uso dessas informações. Informações sobre a tabela, consultar o capítulo Configuração – Parâmetros do Projeto.

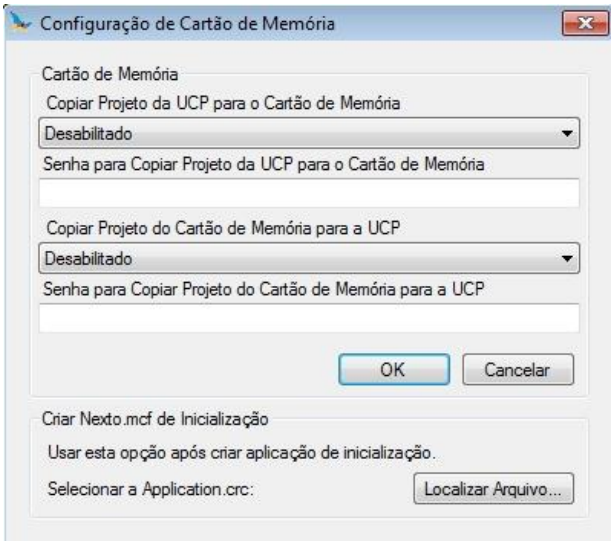
Configuração de Cartão de Memória
Cartão de Memória
Copiar Projeto da UCP para o Cartão de Memória
Desabilitado
Senha para Copiar Projeto da UCP para o Cartão de Memória
Copiar Projeto do Cartão de Memória para a UCP
Desabilitado
Senha para Copiar Projeto do Cartão de Memória para a UCP
OK
Cancelar
Criar Nexto.mcf de Inicialização
Usar esta opção após criar aplicação de inicialização.
Selecionar a Application.crc:
Localizar Arquivo...

Figura 4-66. Configurações do Cartão de Memória

Quando uma senha é configurada para o cartão de memória no MasterTool, é necessário executar os passos a seguir para que, ao enviar o projeto, o arquivo criptografado que é gerado pelo MasterTool tenha a senha incluída no seu conteúdo e também seja enviado.

Primeiramente, configurar a(s) senha(s), apenas com caracteres numéricos, e clicar no botão “OK”, para que esta(s) seja(m) registrada(s). A utilização de senha não é obrigatória. Caso não se queira utilizar senha, o procedimento inicia a partir do próximo passo, que é executar no menu Comunicação o comando “Criar Aplicação de Inicialização”, lembrando que não se pode estar logado na UCP para realizar esse procedimento. Após executar esse comando, dois arquivos são criados. Um com a extensão “app” e outro com a extensão “crc”. Para concluir a operação de configuração da(s) senha(s) é necessário clicar novamente no botão “Cartão de Memória”, que está na configuração dos Parâmetros Gerais da UCP e então localizar o arquivo com extensão “crc” gerado no passo anterior, utilizando o botão “Localizar Arquivo...”. Executados esses passos, o MasterTool IEC XE irá enviar todos os arquivos necessários para realizar as operações de envio e recebimento de projetos via cartão de memória.

Caso o cartão esteja montado, a senha será gravada no mesmo. Caso contrário, a senha configurada no MasterTool será solicitada se o usuário tentar transferir o projeto da UCP para o cartão.

Para realizar o envio da UCP para o cartão de memória ou vice-versa, o usuário, além de habilitar no software MasterTool IEC XE e colocar a senha, terá que acessar o Menu “Cartão de Memória” na UCP, utilizando o botão de diagnósticos, e selecionar a opção de transferência desejada. Após, será solicitada a senha, caso o usuário tenha configurado durante a configuração da aplicação. Então, com um pressionamento curto no botão de diagnósticos os dígitos são incrementados e com um pressionamento longo são confirmados. No sexto dígito confirmado, a UCP irá consistir a senha e iniciará o processo. Quando as senhas, tanto da aplicação que está na UCP quanto da aplicação que está no cartão de memória, forem iguais, não é requisitado a inserção das senhas no menu da UCP para realizar as transferências das aplicações. Maiores informações sobre a utilização do botão de diagnósticos, consultar capítulo One Touch Diag.

Para remover o cartão de memória, basta pressionar a tecla MS com um pressionamento longo, e aguardar até que o ícone do cartão desapareça da tela de status do visor gráfico.

ATENÇÃO: Caso exista algum arquivo na raiz do cartão de memória nomeado “NextoMemCard” ou “Backup”, o mesmo será excluído para a criação das pastas de mesmo nome, utilizadas pela UCP para armazenamento do projeto aplicativo e do Project archive. Pastas com estes nomes não serão sobrescritas.

Acesso no MasterTool

O acesso à memória do cartão está vinculado à mesma tela de memória de usuário no software MasterTool IEC XE, sendo ele montado na pasta “MemoryCard”. Os diretórios “NextoMemCard” e “Backup" são criados dentro do cartão de memória toda a vez que ele for inserido na UCP. Caso eles já existam, o sistema irá reconhecer e não sobrescreverá as pastas.

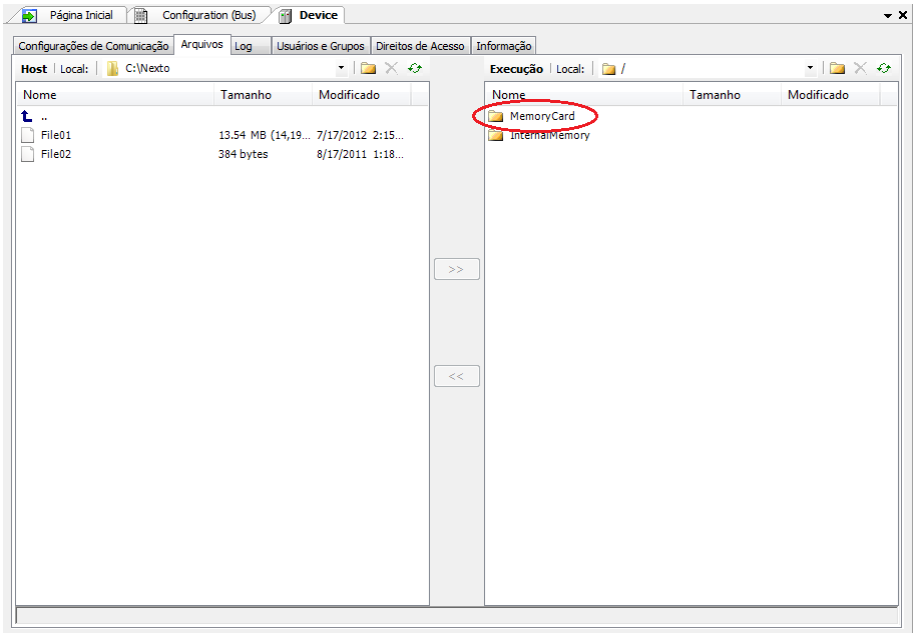

Figura 4-67. Diretório Raiz com Cartão de Memória Inserido

A transferência de arquivos ocorre de maneira semelhante à utilização da memória de usuário(Memória de Arquivos de Usuário), basta acessar o diretório “MemoryCard” e enviar os arquivos, conforme mostra a Figura 4-68.

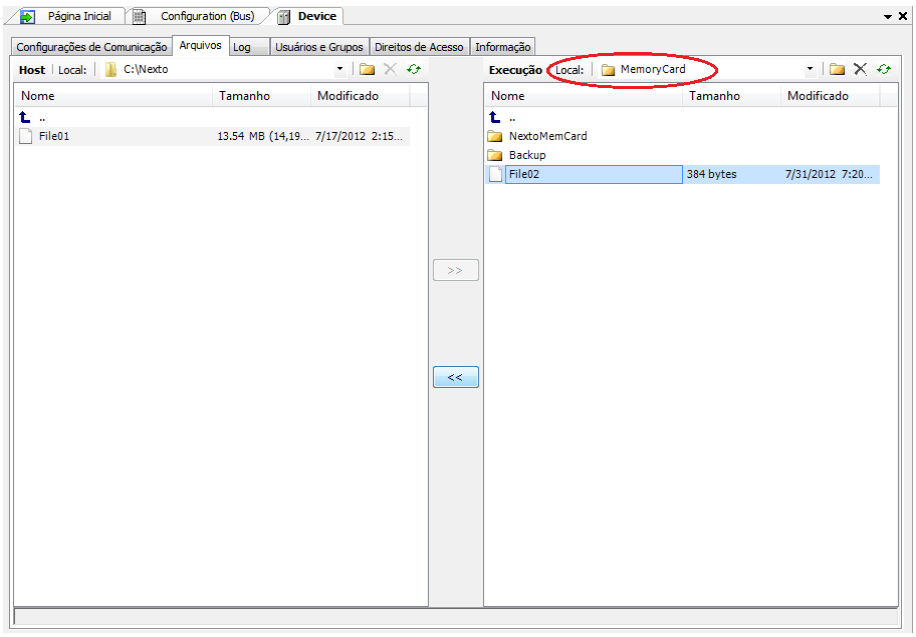

Figura 4-68. Arquivos Salvos no Cartão de Memória

Dentro do diretório do cartão de memória, além dos arquivos que estiverem armazenados no cartão, estarão as pastas “NextoMemCard” e “Backup”. Nestas pastas ficarão salvos o aplicativo e o projeto atual caso o usuário escolha transferi-los ou fazer um backup dos mesmos através do menu da UCP.

ATENÇÃO:
O tempo de transferência de arquivos depende da diferença do tempo de intervalo menos o tempo de execução médio da(s) tarefa(s) em execução (tempo disponível até o próximo ciclo da tarefa), isto é, quanto maior for essa diferença para cada tarefa em uma aplicação, mais rápida deverá ser a transferência de um dado a partir do cartão de memória para a UCP/MasterTool IEC XE ou vice-versa.
A transferência de arquivos para o cartão de memória será mais lenta que a transferência para a memória interna da UCP. Para uma UCP em Modo Stop ou sem nenhuma aplicação, a taxa de transferência se aproxima de 100 Kbytes/s.

Menu Informativo e de Configuração da UCP

O acesso ao Menu Informativo e de Configuração das UCPs Nexto, assim como o acesso detalhado aos diagnósticos, estão disponíveis através de níveis, sendo que para acessar as informações do menu, trocar de nível e modificar alguma configuração, basta dar um pressionamento longo no botão de diagnóstico e, para navegar pelos itens de mesmo nível, basta dar um pressionamento curto no botão de diagnóstico. Consultar o capítulo One Touch Diag para verificar o funcionamento e a diferença entre tipos de pressionamento no botão de diagnósticos.

A Tabela 4-91 mostra os níveis do menu e o tipo de cada tela disponível na UCP, ou seja, se ela é de caráter informativo, configurável ou se retorna um nível.

Nível 1	Nível 2	Nível 3	Tipo
HARDWARE	TEMPERATURA	-	Informativo
	CONTRASTE	NIVEL CONTRASTE	Configurável
	DATA E HORA	-	Informativo
	VOLTAR	-	Retorna Nível
IDIOMAS	ENGLISH	>ENGLISH	Configurável
	PORTUGUES	>PORTUGUES	Configurável
	ESPANOL	>ESPANOL	Configurável
	VOLTAR	-	Retorna Nível
REDE	END. IP NET 1	-	Informativo
	MASCARA NET 1		Informativo
	END. IP NET 2		Informativo
	MASCARA NET 2		Informativo
	VOLTAR		Retorna Nível
REDUNDANCIA	IDENT. CP	-	Informativo
	ESTADO REM.		Informativo
	SINCR. PROJ.		Informativo
	VOLTAR		Retorna Nível
SOFTWARE	FIRMWARE	-	Informativo
	BOOTLOADER		Informativo
	PROC. AUX.		Informativo
	VOLTAR		Retorna Nível
CARTAO DE MEM.	CARTAO > UCP	SENHA UCP	Configurável
	UCP > CARTAO	SENHA CM	Configurável
	FORMATAR	CONFIRMA ?	Configurável
	VOLTAR	-	Retorna Nível
VOLTAR	-	-	Retorna Nível

Tabela 4-91. Níveis do Menu da UCP

Notas:

Cartão de Memória: O cartão de memória somente estará disponível no menu caso o mesmo esteja conectado na UCP Nexto.

Senha: A senha para acesso aos dados do cartão de memória somente será necessária caso ela seja configurada no software MasterTool IEC XE. Não é possível editar a senha via menu.

Rede: Os itens correspondentes à interface NET 2 somente estão disponíveis nas UCPs NX3020 e NX3030.

Redundância: O menu "Redundancia" somente estará disponível caso a UCP NX3030 esteja identificada como Redundante.

Conforme já mostrou a Tabela 4-91, entre as opções disponíveis para visualização e alteração, encontram-se os principais dados necessários ao usuário, como:

- Informações sobre os recursos de hardware:
  - TEMPERATURA – Temperatura interna da UCP (Ex.: 36 C 97 F)
  - CONTRASTE – Ajuste do contraste do visor frontal da UCP
  - DATA E HORA – Data e hora configuradas na UCP (Ex.: 2001.01.31 00:00)
- Alteração do idioma do menu da UCP:
  - PORTUGUES – Altera o idioma para Português
  - ENGLISH – Altera o idioma para Inglês
  - ESPANOL – Altera o idioma para Espanhol
- Visualização de informações sobre a rede configurada no dispositivo:
  - END. IP NET 1 – Endereço IP (Ex.: 192.168.0.1)
  - MASCARA NET 1 – Máscara de subrede (Ex.: 255.255.255.0)
  - END. IP NET 2 – Endereço IP (Ex.: 192.168.0.2)
  - MASCARA NET 2 – Máscara de subrede (Ex.: 255.255.255.0)
- Informações sobre as versões de software:
  - FIRMWARE – Versão de software da UCP (Ex.: 1.0.0.0)
  - BOOTLOADER – Versão do bootloader da UCP (Ex.: 1.0.0.0)
  - PROC. AUX. – Versão do processador auxiliar da UCP (Ex.: 1.0.0.0)
- Acesso às informações de redundância do CP:
  - IDENT. CLP – Informa a identificação do CP na redundância. Possíveis informações:
    - CPA
    - CPB
- ESTADO REM. – Informa o estado do CP redundante remoto. Possíveis estados:
  - ATIVO
  - RESERVA
  - INATIVO
  - NAO CONFIG.
  - INICIANDO
  - INDISPON.
- SINCR. PROJ. – Informa se a sincronização de projetos está habilitada
  - CONECTADO
  - NÃO CONEC.
  - DESABILITADO
  - SINC. INI.
  - SINCRONIZADO
- Acesso aos dados do Cartão de Memória:
  - CARTAO>UCP – Transferência do projeto do cartão de memória para a UCP
  - UCP>CARTAO – Transferência do projeto da UCP para o cartão de memória
  - FORMATAR – Formata o cartão para o sistema de arquivos FAT32

A Figura 4-69 descreve um exemplo de como operar o menu das UCPs Nexto, através do procedimento de ajuste do contraste a partir da tela de Status. Além de facilitar a configuração, é possível identificar todos os níveis de tela e o tipo de pressionamento para navegar entre as mesmas, sendo que para modificar os outros parâmetros, como Idioma e inserir a(s) senha(s) no Cartão de Memória, basta seguir a mesma lógica de acesso. O pressionamento curto mostra que o contraste está sendo incrementado (mais claro), sendo que no próximo pressionamento após o seu valor máximo, ele retorna ao valor mínimo (menos claro). O pressionamento longo mostra a confirmação do contraste desejado e o retorno ao nível anterior.

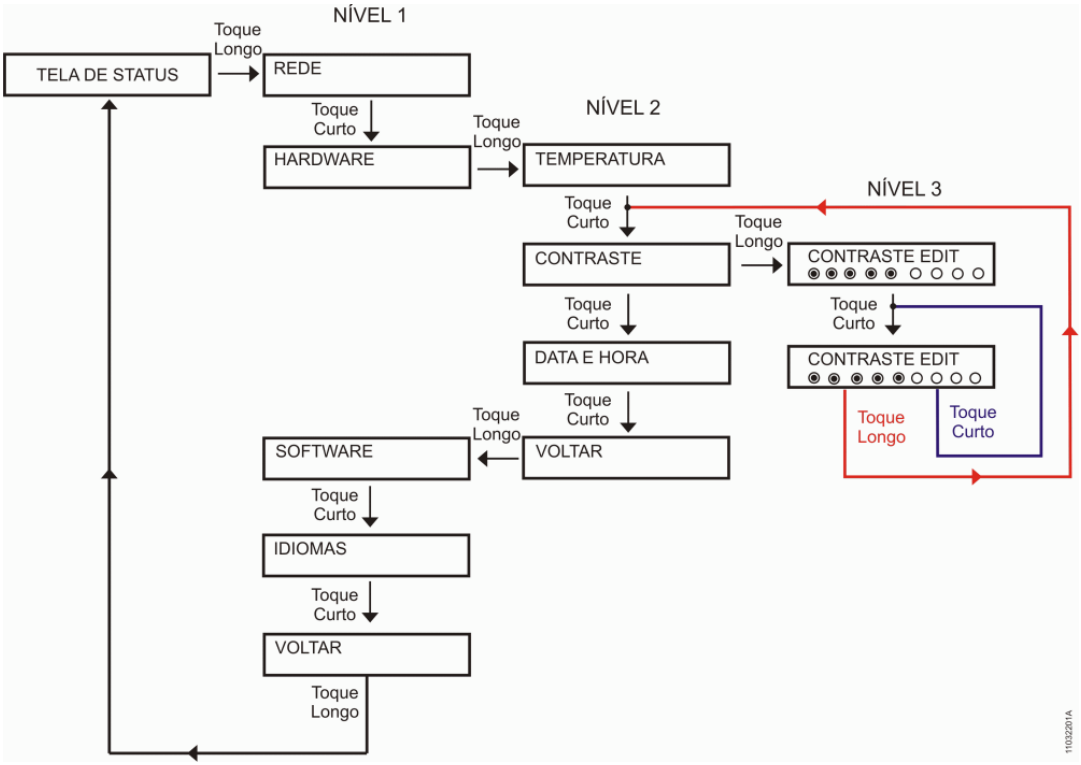

Figura 4-69. Ajuste do Contraste

Além do menu das UCPs Nexto ser encerrado através de um pressionamento longo no botão de diagnósticos na tela VOLTAR do nível 1, também existem outras condições de saída, as quais estão descritas abaixo:

- Toque curto, em qualquer momento, em um outro módulo existente no barramento, faz com que o menu passe a descrever o diagnóstico dele.
- Tempo de inatividade, em qualquer nível, superior a 5 s.

Blocos Funcionais e Funções

Blocos Funcionais Especiais para Comunicação Serial

Os blocos funcionais especiais para comunicação serial possibilitam o acesso local (COM 1 e COM 2) e também a portas seriais remotas (módulos de expansão). Dessa forma, o usuário poderá criar seus próprios protocolos e manipular as portas seriais como quiser, seguindo as linguagens da IEC 61131-3 disponíveis no software MasterTool IEC XE. Os blocos estão disponíveis dentro da biblioteca NextoSerial, a qual deve ser adicionada ao projeto para que seja possível utilizá-los (para realizar o procedimento de inserção de uma biblioteca, consultar o Manual de Programação IEC 61131 – MP399048, capítulo Bibliotecas).

Os blocos funcionais especiais para comunicação serial podem levar vários ciclos (consecutivas chamadas) para completar a execução da tarefa. Às vezes, um bloco pode completar em um único ciclo, mas, em geral, necessita de vários ciclos. A execução da tarefa associada a um bloco pode

compreender vários passos, sendo que alguns dependem de eventos externos, os quais podem ter atrasos significantes para o sistema. O bloco funcional não pode implementar rotinas para ocupar o tempo, enquanto aguarda por esses eventos, pois assim iria utilizar todo recurso da UCP. A solução poderia ser a criação de blocos funcionais bloqueadores, mas isso não é aconselhável pois iria complicar a aplicação do usuário, pois normalmente não se tem disponível a programação multitarefa. Então, quando um evento externo é esperado, os blocos funcionais da serial são finalizados e o controle é retornado para o programa de chamada. O tratamento da tarefa continua no próximo ciclo, ou seja, na próxima vez que o bloco for chamado.

Antes de descrever os blocos funcionais especiais para controle das interfaces seriais, é importante conhecer os Datatypes, ou seja, os tipos de dados utilizados pelos blocos:

Datatype	Opção	Descrição
SERIAL_BAUDRATE	BAUD200	Lista todas as possibilidades da taxa de transmissão (bits por segundo)
	BAUD300	
	BAUD600	
	BAUD1200	
	BAUD1800	
	BAUD2400	
	BAUD4800	
	BAUD9600	
	BAUD19200	
	BAUD38400	
	BAUD57600	
	BAUD115200	
SERIAL_DATABITS	DATABITS_5	Lista todas as possibilidades de bits de dado.
	DATABITS_6	
	DATABITS_7	
	DATABITS_8	
SERIAL_HANDSHAKE	Define todas as possibilidades dos sinais de modem para as configurações:	
	RS232_RTS	Controla a porta RS-232C da UCP Nexto. O RTS é habilitado no início da transmissão e reiniciado assim que possível após o final da transmissão. Por exemplo, pode ser utilizado para controlar um conversor RS-232/RS-485 externo.
	RS232_RTS_OFF	Controla a porta RS-232C da UCP Nexto. O sinal RTS está sempre desligado.
	RS232_RTS_ON	Controla a porta RS-232C da UCP Nexto. O sinal RTS está sempre ligado.
	RS232_RTS_CTS	Controla a porta RS-232C da UCP Nexto. Caso o CTS esteja desabilitado, o RTS é habilitado. Então, aguarda-se o CTS ser habilitado para a transmissão começar e o RTS é reiniciado, o mais rápido possível, no final da transmissão. Ex: Controle de rádio modems com o mesmo sinal de modem.
	RS232_MANUAL	Controla a porta RS-232C da UCP Nexto. O usuário é responsáel por controlar todos os sinais (RTS, DTR, CTS, DSR, DCD).
SERIAL_MODE	NORMAL_MODE	Modo normal de operação da comunicação serial.
	EXTENDED_MODE	Modo estendido de operação da comunicação serial, no qual são fornecidas informações sobre o frame de dados recebido.

SERIAL_PARAMETERS	Define todos os parâmetros de configuração da porta serial:	
	BAUDRATE	Definido em SERIAL_BAUDRATE.
	DATABITS	Definido em SERIAL_DATABITS.
	STOPBITS	Definido em SERIAL_STOPBITS.
	PARITY	Definido em SERIAL_PARITY.
	HANDSHAKE	Definido em SERIAL_HANDSHAKE.
	UART_RX_THRESHOLD	Quantidade de bytes que devem ser recebidos para gerar uma nova interrupção na UART. Valores baixos fazem o TIMESTAMP mais preciso quando o MODO ESTENDIDO é utilizado e minimiza os erros de overrun. No entanto, valores baixos podem causar muitas interrupções que podem retardar a UCP.
	MODE	Definido em SERIAL_MODE.
	ENABLE_RX_ON_TX	Quando verdadeiro, todos os bytes recebidos durante a transmissão serão descartados em vez de ir para a fila de RX. Utilizado para desabilitar a operação full-duplex na interface RS-422.
	ENABLE_DCD_EVENT	Quando verdadeiro, gera um evento externo quando o DCD é alterado.
	ENABLE_CTS_EVENT	Quando verdadeiro, gera um evento externo quando o CTS é alterado.
SERIAL_PARITY	PARITY_NONE	Lista todas as possibilidades de paridade.
	PARITY_ODD	
	PARITY_EVEN	
	PARITY_MARK	
	PARITY_SPACE	
SERIAL_PORT	COM 1	Lista todas as portas seriais disponíveis (COM 10, COM 11, COM 12, COM 13, COM 14, COM 15, COM 16, COM 17, COM 18, COM 19 – módulos de expansão).
	COM 2	
SERIAL_RX_CHAR_EXTENDED	Define um caractere da fila RX modo estendido.	
	RX_CHAR	Byte de dados.
	RX_ERROR	Código de erro.
	RX_TIMESTAMP	Silêncio devido ao caractere anterior ou devido a outro evento que aconteceu antes desse caracterere (configuração da porta serial, final da transmissão).
SERIAL_RX_QUEUE_STATUS	Contém alguns campos que disponibilizam informações de status/erro sobre a fila RX, utilizados quando é utilizado o formato normal (sem erro e informações de timestamping):	
	RX_FRAMING_ERRORS	Contador de erros de frame, ou seja, formação incorreta do caractere – falta de bit de parada, taxa de transmissão incorreto, entre outros – desde a configuração da porta serial. Retorna para zero caso atinja o valor máximo (65535).
	RX_PARITY_ERRORS	Contador de erros de paridade desde a configuração da porta serial. Retorna para zero caso atinja o valor máximo (65535).
	RX_BREAK_ERRORS	Contador de erros de interrupção desde a configuração da porta serial, ou seja, linha ativa maior do

		que o tempo de um caractere. Retorna para zero caso atinja o valor máximo (65535).
	RX_FIFO_OVERRUN_ERRORS	Contador de erros de overrun na FIFO RX desde a configuração da porta serial, ou seja, erro no threshold configurado para a FIFO RX. Retorna para zero caso atinja o valor máximo (65535).
	RX_QUEUE_OVERRUN_ERRORS	Contador de erros de overrun na fila RX desde a configuração da porta serial, ou seja, o valor máximo de caracteres (1024) foi ultrapassado e os dados estão sendo sobreescritos. Retorna zero caso atinja o valor máximo (65535).
	RX_ANY_ERRORS	Soma dos 5 últimos contadores de erros (frame, paridade, interrupção, overrun RX FIFO, overrun fila RX).
	RX_REMAINING	Número de caracteres remanescentes na fila RX.
SERIAL_STATUS	Lista os códigos de erro críticos que podem ser retornados pelos blocos funcionais da serial. Cada bloco retorna erros específicos, os quais serão mencionados na descrição dos mesmos:	
	NO_ERROR	Não existem erros.
	ILLEGAL_*	Retorna os parâmetros com valores inválidos ou fora da faixa: - SERIAL_PORT - SERIAL_MODE - BAUDRATE - DATA_BITS - PARITY - STOP_BITS - HANDSHAKE - UART_RX_THRESHOLD - TIMEOUT - TX_BUFF_LENGTH - HANDSHAKE_METHOD - RX_BUFF_LENGTH
	PORT_BUSY	Indica que a porta serial está sendo utilizada por outra instância
	HW_ERROR_UART	Erro de hardware detectado na UART.
	HW_ERROR_REMOTE	Erro de hardware ao comunicar com a porta serial remota.
	CTS_TIMEOUT_ON	Time-out na espera do CTS ser habilitado, no sinais de modem RS-232C RTS/CTS, no bloco SERIAL_TX.
	CTS_TIMEOUT_OFF	Time-out na espera do CTS ser desabilitado, no handshake RS-232C RTS/CTS, no bloco SERIAL_TX.
	TX_TIMEOUT_ERROR	Time-out na espera pelo final da transmissão no bloco SERIAL_TX.
	RX_TIMEOUT_ERROR	Time-out na espera de todos os caracteres no bloco SERIAL_RX ou SERIAL_RX_EXTENDED.
	FB_SET_CTRL_NOT_ALLOWED	O bloco SET_CTRL não pode ser utilizado caso o sinal de modem seja diferente de RS232_MANUAL.
	FB_GET_CTRL_NOT_ALLOWED	O bloco GET_CTRL não pode ser utilizado caso o sinal de modem seja diferente de RS232_MANUAL.
	FB_SERIAL_RX_NOT_ALLOWED	O bloco SERIAL_RX não está disponível para a fila RX, modo

		estendido.
	FB_SERIAL_RX_EXTENDED_NOT_ALLOWED	O bloco SERIAL_RX_EXTENDED não está disponível para a fila RX, modo normal.
	DCD_INTERRUPT_NOT_ALLOWED	A interrupção pelo sinal DCD não pode ser habilitado caso a porta serial não possua o respectivo pino.
	CTS_INTERRUPT_NOT_ALLOWED	A interrupção pelo sinal CTS não pode ser habilitada caso o sinal de modem seja diferente de RS232_MANUAL ou caso a porta serial não possua o respectivo pino.
	DSR_INTERRUPT_NOT_ALLOWED	A interrupção pelo sinal DSR não pode ser habilitado caso a porta serial não possua o respectivo pino. (As UCPs Nexto não possuem esse sinal nas portas locais)
	NOT_CONFIGURED	O bloco funcional não pode ser utilizado antes da porta serial ser configurada.
	INTERNAL_ERROR	Indica que algum problema interno ocorreu na porta serial.
SERIAL_STOPBITS	STOPBITS_1	Lista todas as possibilidades de bits de parada.
	STOPBITS_2	
	STOPBITS_1_5	

Tabela 4-92. Datatypes Blocos Funcionais Serial

SERIAL_CFG

Esse bloco funcional é utilizado para configurar e inicializar a porta serial desejada. Após a chamada do bloco, todas as filas RX e TX associadas à porta serial e os FIFOs RX e TX, são reiniciados.

Figura 4-70. Bloco de Configuração da Serial

Parâmetros de entrada	Tipo	Descrição
REQUEST	BOOL	Essa variável, quando verdadeira, habilita o uso do bloco funcional.
PORT	SERIAL_PORT	Seleciona a porta serial, conforme descrito no datatype SERIAL_PORT.
PARAMETERS	SERIAL_PARAMETERS	Essa estrutura define os parâmetros de configuração da porta serial, conforme descrito no datatype SERIAL_PARAMETERS.

Tabela 4-93. Parâmetros de Entrada Serial_CFG

Parâmetros de saída	Tipo	Descrição
DONE	BOOL	Essa variável é verdadeira quando o bloco funcional for executado por completo, caso contrário, ela é falsa.
EXEC	BOOL	Essa variável é verdadeira enquanto o bloco funcional estiver sendo executado, caso contrário, ela é falsa.
ERROR	BOOL	Essa variável é verdadeira quando o bloco funcional concluir a sua execução com algum

		erro, caso contrário, ela é falsa. Está vinculada à variável DONE, pois seu estado é exibido após a conclusão do bloco funcional.
STATUS	SERIAL_STATUS	Caso a variável ERROR seja verdadeira, a estrutura STATUS exibirá o erro encontrado na execução do bloco funcional. Os estados, já descritos no datatype SERIAL_STATUS, possíveis são: - NO_ERROR, - ILLEGAL_SERIAL_PORT, - ILLEGAL_SERIAL_MODE, - ILLEGAL_BAUDRATE, - ILLEGAL_DATA_BITS, - ILLEGAL_PARITY, - ILLEGAL_STOP_BITS, - ILLEGAL_HANDSHAKE, - ILLEGAL_UART_RX_THRESHOLD, - PORT_BUSY, - HW_ERROR_UART, - HW_ERROR_REMOTE, - DCD_INTERRUPT_NOT_ALLOWED, - CTS_INTERRUPT_NOT_ALLOWED, - DSR_INTERRUPT_NOT_ALLOWED.

Tabela 4-94. Parâmetros de Saída Serial_CFG

Exemplo de utilização em Linguagem ST, após a biblioteca Nexto Serial ter sido inserida no projeto:

```
PROGRAM MainPrg
VAR
   Config: SERIAL_CFG;
   Port: SERIAL_PORT := COM1;
   Parameters: SERIAL_PARAMETERS := (BAUDRATE := BAUD9600,
      DATABITS := DATABITS_8,
      STOPBITS := STOPBITS_1,
      PARITY := PARITY_NONE,
      HANDSHAKE := RS232_RTS,
      UART_RX_THRESHOLD := 8,
      MODE :=NORMAL_MODE,
      ENABLE_RX_ON_TX := FALSE,
      ENABLE_DCD_EVENT := FALSE,
      ENABLE_CTS_EVENT := FALSE);
   Status: SERIAL_STATUS;
END_VAR
//ENTRADAS:
Config.REQUEST := TRUE;
Config.PORT := Port;
Config.PARAMETERS := Parameters;
//FUNÇÃO:
Config();
//SAÍDAS:
Config.DONE;
Config.EXEC;
Config.ERROR;
Status := Config.STATUS;         //Caso seja necessário tratar o erro.
```

SERIAL_GET_CFG

Esse bloco funcional é utilizado para capturar as configurações da porta serial desejada.

Figura 4-71. Bloco para Capturar a Configuração da Serial

Parâmetros de entrada	Tipo	Descrição
REQUEST	BOOL	Essa variável, quando verdadeira, habilita o uso do bloco funcional.
PORT	SERIAL_PORT	Seleciona a porta serial, conforme descrito no datatype SERIAL_PORT.

Tabela 4-95. Parâmetros de Entrada Serial_GET_CFG

Parâmetros de saída	Tipo	Descrição
DONE	BOOL	Essa variável é verdadeira quando o bloco funcional for executado por completo, caso contrário, ela é falsa.
EXEC	BOOL	Essa variável é verdadeira enquanto o bloco funcional estiver sendo executado, caso contrário, ela é falsa.
ERROR	BOOL	Essa variável é verdadeira quando o bloco funcional concluir a sua execução com algum erro, caso contrário, ela é falsa. Está vinculada à variável DONE, pois seu estado é exibido após a conclusão do bloco funcional.
STATUS	SERIAL_STATUS	Caso a variável ERROR seja verdadeira, a estrutura STATUS exibirá o erro encontrado na execução do bloco. Os estados, já descritos no datatype SERIAL_STATUS, possíveis são: - NO_ERROR - ILLEGAL_SERIAL_PORT - PORT_BUSY - HW_ERROR_UART - HW_ERROR_REMOTE - NOT_CONFIGURED
PARAMETERS	SERIAL_PARAMETERS	Essa estrutura recebe os parâmetros de configuração da porta serial desejada, conforme descrito no datatype SERIAL_PARAMETERS.

Tabela 4-96. Parâmetros de Saída Serial_GET_CFG

Exemplo de utilização em Linguagem ST, após a biblioteca ser inserida no projeto:

```
PROGRAM MainPrg
VAR
   GetConfig: SERIAL_GET_CFG;
   Port: SERIAL_PORT := COM1;
   Parameters: SERIAL_PARAMETERS;
   Status: SERIAL_STATUS;
END_VAR
//ENTRADAS:
GetConfig.REQUEST := TRUE;
GetConfig.PORT := Port;
//FUNÇÃO:
GetConfig();
```

```
//SAÍDAS:
GetConfig.DONE;
GetConfig.EXEC;
GetConfig.ERROR;
Status := GetConfig.STATUS;      //Caso seja necessário tratar o erro.
Parameters := GetConfig.PARAMETERS;    //Recebe os parâmetros da porta
serial desejada.
```

SERIAL_GET_CTRL

Esse bloco funcional é utilizado para ler os sinais de controle CTS, DSR e DCD, caso eles estejam disponíveis na porta serial. Será retornado um valor falso quando os sinais de controle não existirem.

Figura 4-72. Bloco para Visualizar os Sinais de Controle

Parâmetros de entrada	Tipo	Descrição
REQUEST	BOOL	Essa variável, quando verdadeira, habilita o uso do bloco funcional.
PORT	SERIAL_PORT	Seleciona a porta serial, conforme descrito no datatype SERIAL_PORT.

Tabela 4-97. Parâmetros de Entrada Serial_GET_CTRL

Parâmetros de saída	Tipo	Descrição
DONE	BOOL	Essa variável é verdadeira quando o bloco funcional for executado por completo, caso contrário, ela é falsa.
EXEC	BOOL	Essa variável é verdadeira enquanto o bloco funcional estiver sendo executado, caso contrário, ela é falsa.
ERROR	BOOL	Essa variável é verdadeira quando o bloco funcional concluir a sua execução com algum erro, caso contrário, ela é falsa. Está vinculada à variável DONE, pois seu estado é exibido após a conclusão do bloco funcional.
STATUS	SERIAL_STATUS	Caso a variável ERROR seja verdadeira, a estrutura STATUS exibirá o erro encontrado na execução do bloco. Os status, já descritos no datatype SERIAL_STATUS, possíveis são: - NO_ERROR - ILLEGAL_SERIAL_PORT - PORT_BUSY - HW_ERROR_UART - HW_ERROR_REMOTE - FB_GET_CTRL_NOT_ALLOWED - NOT_CONFIGURED
CTS_VALUE	BOOL	Valor lido no sinal de controle CTS.
DSR_VALUE	BOOL	Valor lido no sinal de controle DSR.
DCD_VALUE	BOOL	Valor lido no sinal de controle DCD.

Tabela 4-98. Parâmetros de Saída Serial_GET_CTRL

Exemplo de utilização em Linguagem ST, após a biblioteca ser inserida no projeto e a porta serial ser configurada:

```
PROGRAM MainPrg
VAR
   Get_Control: SERIAL_GET_CTRL;
   Port: SERIAL_PORT := COM1;
   Status: SERIAL_STATUS;
END_VAR
//ENTRADAS:
Get_Control.REQUEST := TRUE;
Get_Control.PORT := Port;
//FUNÇÃO:
Get_Control();
//SAÍDAS:
Get_Control.DONE;
Get_Control.EXEC;
Get_Control.ERROR;
Status := Get_Control.STATUS;          //Caso seja necessário tratar o
erro.
Get_Control.CTS_VALUE;
Get_Control.DSR_VALUE;
Get_Control.DCD_VALUE;
```

SERIAL_GET_RX_QUEUE_STATUS

Esse bloco funcional é utilizado para ler algumas informações de status sobre a fila RX, sendo especialmente desenvolvido para o modo normal, mas pode também ser utilizado no modo estendido.

Figura 4-73. Bloco para Visualizar o Status da Fila RX

Parâmetros de entrada	Tipo	Descrição
REQUEST	BOOL	Essa variável, quando verdadeira, habilita o uso do bloco funcional.
PORT	SERIAL_PORT	Seleciona a porta serial, conforme descrito no datatype SERIAL_PORT.

Tabela 4-99. Parâmetros de Entrada Serial_GET_RX_QUEUE_STATUS

Parâmetros de saída	Tipo	Descrição
DONE	BOOL	Essa variável é verdadeira quando o bloco funcional for executado por completo, caso contrário, ela é falsa.
EXEC	BOOL	Essa variável é verdadeira enquanto o bloco funcional estiver sendo executado, caso contrário, ela é falsa.
ERROR	BOOL	Essa variável é verdadeira quando o bloco funcional concluir a sua execução com algum erro, caso contrário, ela é falsa. Está vinculada à variável DONE, pois seu estado é exibido após a conclusão do bloco de função.
STATUS	SERIAL_STATUS	Caso a variável ERROR seja verdadeira, a estrutura STATUS exibirá o erro encontrado na execução do bloco funcional. Os status, já descritos no datatype SERIAL_STATUS, possíveis são: - NO_ERROR - ILLEGAL_SERIAL_PORT - PORT_BUSY - HW_ERROR_UART

		- HW_ERROR_REMOTE - NOT_CONFIGURED
RXQ_STATUS	SERIAL_RX_QUEUE_STATUS	Retorna status/erros da fila RX, conforme descrito no datatype. SERIAL_RX_QUEUE_STATUS.

Tabela 4-100. Parâmetros de Saída Serial_GET_RX_QUEUE_STATUS

Exemplo de utilização em Linguagem ST, após a biblioteca ser inserida no projeto e a porta serial ser configurada:

```
PROGRAM MainPrg
VAR
    Get_Status: SERIAL_GET_RX_QUEUE_STATUS;
    Port: SERIAL_PORT := COM1;
    Status: SERIAL_STATUS;
    Status_RX: SERIAL_RX_QUEUE_STATUS;
END_VAR
//ENTRADAS:
Get_Status.REQUEST := TRUE;
Get_Status.PORT := Port;
//FUNÇÃO:
Get_Status();
//SAÍDAS:
Get_Status.DONE;
Get_Status.EXEC;
Get_Status.ERROR;
Status := Get_Status.STATUS;           //Caso seja necessário tratar o
erro.
Status_RX := Get_Status.RXQ_STATUS;    //Caso seja necessário tratar o
erro da fila RX.
```

SERIAL_PURGE_RX_QUEUE

Esse bloco funcional é utilizado para limpar a fila RX, local e remota, da porta serial. A UART RX FIFOs também é reiniciada.

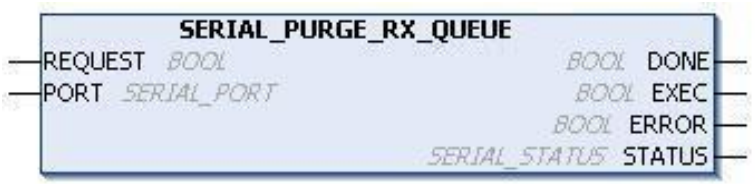

Figura 4-74. Bloco para Limpar a Fila RX

Parâmetros de entrada	Tipo	Descrição
REQUEST	BOOL	Essa variável, quando verdadeira, habilita o uso do bloco funcional.
PORT	SERIAL_PORT	Seleciona a porta serial, conforme descrito no datatype SERIAL_PORT.

Tabela 4-101. Parâmetros de Entrada Serial_PURGE_RX_QUEUE

Parâmetros de saída	Tipo	Descrição
DONE	BOOL	Essa variável é verdadeira quando o bloco funcional for executado por completo, caso contrário, ela é falsa.
EXEC	BOOL	Essa variável é verdadeira enquanto o bloco funcional estiver sendo executado, caso contrário, ela é falsa.
ERROR	BOOL	Essa variável é verdadeira quando o bloco funcional

		concluir a sua execução com algum erro, caso contrário, ela é falsa. Está vinculada à variável DONE, pois seu estado é exibido após a conclusão do bloco funcional .
STATUS	SERIAL_STATUS	Caso a variável ERROR seja verdadeira, a estrutura STATUS exibirá o erro encontrado na execução do bloco funcional. Os status, já descritos no datatype SERIAL_STATUS, possíveis são: - NO_ERROR - ILLEGAL_SERIAL_PORT - PORT_BUSY - HW_ERROR_UART - HW_ERROR_REMOTE - NOT_CONFIGURED

Tabela 4-102. Parâmetros de Saída Serial_PURGE_RX_QUEUE

Exemplo de utilização em Linguagem ST, após a biblioteca ser inserida no projeto e a porta serial ser configurada:

```
PROGRAM MainPrg
VAR
   Purge_Queue: SERIAL_PURGE_RX_QUEUE;
   Port: SERIAL_PORT := COM1;
   Status: SERIAL_STATUS;
END_VAR
//ENTRADAS:
Purge_Queue.REQUEST := TRUE;
Purge_Queue.PORT := Port;
//FUNÇÃO:
Purge_Queue();
//SAÍDAS:
Purge_Queue.DONE;
Purge_Queue.EXEC;
Purge_Queue.ERROR;
Status := Purge_Queue.STATUS;    //Caso seja necessário tratar o erro.
```

SERIAL_RX

Esse bloco funcional é utilizado para receber um buffer da porta serial utilizando o modo normal da fila RX. Neste modo, cada caractere na fila RX ocupa um único byte que contém o dado recebido, ou seja, armazena 5, 6, 7 ou 8 bits, de acordo com a configuração da interface serial.

Figura 4-75. Bloco para Ler Valores do Buffer de Recepção

Parâmetros de entrada	Tipo	Descrição
REQUEST	BOOL	Essa variável, quando verdadeira, habilita o uso do bloco funcional.
PORT	SERIAL_PORT	Seleciona a porta serial, conforme descrito no datatype SERIAL_PORT.
RX_BUFFER_POINTER	POINTER TO BYTE	Ponteiro de um array de bytes para receber os valores do buffer.
RX_BUFFER_LENGTH	UINT	Especifica o número de caracteres esperados no array de bytes. Caso estejam disponíveis mais bytes do que o esperado, somente a quantidade esperada vai ser lida no array de bytes, sendo que os demais

		serão deixados na fila RX (tamanho máximo igual a 1024 caracteres).
RX_TIMEOUT	UINT	Especifica o time-out para receber a quantidade de caracteres esperados. Caso ele seja menor do que a quantidade de bytes a receber, será indicado RX_TIMEOUT_ERROR no parâmetro de saída STATUS. Quando o valor especificado, em ms, for igual a zero, a função irá retornar os dados presentes no buffer.

Tabela 4-103. Parâmetros de Entrada Serial_RX

Parâmetros de saída	Tipo	Descrição
DONE	BOOL	Essa variável é verdadeira quando o bloco funcional for executado por completo, caso contrário, ela é falsa.
EXEC	BOOL	Essa variável é verdadeira enquanto o bloco funcional estiver sendo executado, caso contrário, ela é falsa.
ERROR	BOOL	Essa variável é verdadeira quando o bloco funcional concluir a sua execução com algum erro, caso contrário, ela é falsa. Está vinculada à variável DONE, pois seu estado é exibido após a conclusão do bloco de função.
STATUS	SERIAL_STATUS	Caso a variável ERROR seja verdadeira, a estrutura STATUS exibirá o erro encontrado na execução do bloco funcional. Os status, já descritos no datatype SERIAL_STATUS, possíveis são: - NO_ERROR - ILLEGAL_SERIAL_PORT - PORT_BUSY - HW_ERROR_UART - HW_ERROR_REMOTE - ILLEGAL_RX_BUFF_LENGTH - RX_TIMEOUT_ERROR - FB_SERIAL_RX_NOT_ALLOWED - NOT_CONFIGURED.
RX_RECEIVED	UINT	Retorna o número de caracteres recebidos. Esse número pode estar entre zero e o valor configurado em RX_BUFFER_LENGTH. Caso seja menor, um erro será indicado pelo bloco funcional.
RX_REMAINING	UINT	Retorna o número de caracteres que ainda estão na fila RX depois que o bloco funcional foi executado.

Tabela 4-104. Parâmetros de Saída Serial_RX

Exemplo de utilização em Linguagem ST, após a biblioteca ser inserida no projeto e a porta serial ser configurada:

```
PROGRAM MainPrg
VAR

   Receive: SERIAL_RX;
   Port: SERIAL_PORT := COM1;
   Buffer_Pointer: ARRAY [0..1023] OF BYTE;       //Tamanho máximo.
   Status: SERIAL_STATUS;
END_VAR
//ENTRADAS:
Receive.REQUEST := TRUE;
Receive.PORT := Port;
Receive.RX_BUFFER_POINTER := ADR(Buffer_Pointer);
Receive.RX_BUFFER_LENGTH := 1024;                  //Tamanho máximo.
Receive.RX_TIMEOUT := 10000;
//FUNÇÃO:
Receive();
```

```
//SAÍDAS:
Receive.DONE;
Receive.EXEC;
Receive.ERROR;
Status := Receive.STATUS;        //Caso seja necessário tratar o erro.
Receive.RX_RECEIVED;
Receive.RX_REMAINING;
```

SERIAL_RX_EXTENDED

Esse bloco funcional é utilizado para receber um buffer da porta serial utilizando o modo estendido da fila RX, conforme detalhado no seção Configuração das Interfaces Seriais.

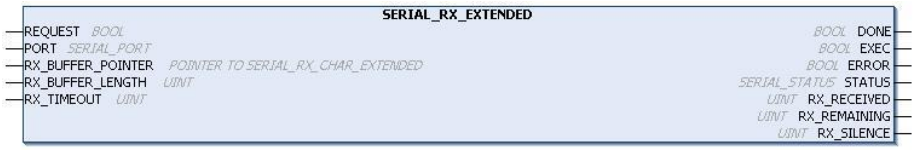

Figura 4-76. Bloco para Leitura do Buffer de Recepção

Parâmetros de entrada	Tipo	Descrição
REQUEST	BOOL	Essa variável, quando verdadeira, habilita o uso do bloco funcional.
PORT	SERIAL_PORT	Seleciona a porta serial, conforme descrito no datatype SERIAL_PORT.
RX_BUFFER_POINTER	POINTER TO SERIAL_RX_CHAR_EXTENDED	Ponteiro de um array de SERIAL_RX_CHAR_EXTENDED para receber os valores do buffer.
RX_BUFFER_LENGTH	UINT	Especifica o número de caracteres esperados no array de SERIAL_RX_CHAR_EXTENDED. Caso estejam disponíveis mais bytes do que o esperado, somente a quantidade esperada vai ser lida no array, sendo que os demais serão deixados na fila RX (tamanho máximo igual a 1024 caracteres).
RX_TIMEOUT	UINT	Especifica o time-out para receber a quantidade de caracteres esperados. Caso ele seja menor do que a quantidade de bytes a receber, será indicado RX_TIMEOUT_ERROR no parâmetro de saída STATUS. Quando o valor especificado, em ms, for igual a zero, a função irá retornar os dados presentes no buffer.

Tabela 4-105. Parâmetros de Entrada Serial_RX_Extended

Parâmetros de saída	Tipo	Descrição
DONE	BOOL	Essa variável é verdadeira quando o bloco funcional for executado por completo, caso contrário, ela é falsa.
EXEC	BOOL	Essa variável é verdadeira enquanto o bloco funcional estiver sendo executado, caso contrário, ela é falsa.
ERROR	BOOL	Essa variável é verdadeira quando o bloco funcional concluir a sua execução com algum erro, caso contrário, ela é falsa. Está vinculada à variável DONE, pois seu estado é exibido após a conclusão do bloco funcional.
STATUS	SERIAL_STATUS	Caso a variável ERROR seja verdadeira, a estrutura STATUS exibirá o erro encontrado na execução do bloco funcional. Os status, já descritos no datatype SERIAL_STATUS, possíveis são: - NO_ERROR - ILLEGAL_SERIAL_PORT - PORT_BUSY - HW_ERROR_UART

		- HW_ERROR_REMOTE - ILLEGAL_RX_BUFF_LENGTH - RX_TIMEOUT_ERROR - FB_SERIAL_RX_EXTENDED_NOT_ALLOWED - NOT_CONFIGURED
RX_RECEIVED	UINT	Retorna o número de caracteres recebidos. Esse número pode estar entre zero e o valor configurado em RX_BUFFER_LENGTH. Caso seja menor, um erro será indicado pelo bloco funcional.
RX_REMAINING	UINT	Retorna o número de caracteres que ainda estão na fila RX depois que o bloco funcional foi executado.
RX_SILENCE	UINT	Retorna o tempo de silêncio na linha RX, medido desde o fim do último caractere recebido. A unidade de tempo é 10 µs. Esse tipo de parâmetro de saída é importante para detectar o tempo de silêncio em protocolos como MODBUS RTU. Pode não ser o tempo de silêncio depois do último caractere recebido por esse bloco funcional, pois somente é verdade se RX_REMANING = 0.

Tabela 4-106. Parâmetros de Saída Serial_RX_Extended

Exemplo de utilização em Linguagem ST, após a biblioteca ser inserida no projeto e a porta serial ser configurada:

```
PROGRAM MainPrg
VAR

   Receive_Ex: SERIAL_RX_EXTENDED;
   Port: SERIAL_PORT := COM1;
   Buffer_Pointer: ARRAY [0..1023] OF SERIAL_RX_CHAR_EXTENDED;
   Status: SERIAL_STATUS;
END_VAR
//ENTRADAS:
Receive_Ex.REQUEST := TRUE;
Receive_Ex.PORT := Port;
Receive_Ex.RX_BUFFER_POINTER := ADR(Buffer_Pointer);
Receive_Ex.RX_BUFFER_LENGTH := 1024;              //Tamanho máximo.
Receive_Ex.RX_TIMEOUT := 10000;
//FUNÇÃO:
Receive_Ex();
//SAÍDAS:
Receive_Ex.DONE;
Receive_Ex.EXEC;
Receive_Ex.ERROR;
Status := Receive_Ex.STATUS;     //Caso seja necessário tratar o erro.
Receive_Ex.RX_RECEIVED;
Receive_Ex.RX_REMAINING;
Receive_Ex.RX_SILENCE;
```

SERIAL_SET_CTRL

Esse bloco funcional é utilizado para escrever nos sinais de controle (RTS e DTR), quando estes estiverem disponíveis na porta serial. Também pode determinar uma condição de ocupado para a transmissão, através do parâmetro BREAK, sendo que somente pode ser utilizado se o sinal de modem estiver configurado para RS232_MANUAL.

Figura 4-77. Bloco para Escrever nos Sinais de Controle

Parâmetros de entrada	Tipo	Descrição
REQUEST	BOOL	Essa variável, quando verdadeira, habilita o uso do bloco funcional.
PORT	SERIAL_PORT	Seleciona a porta serial, conforme descrito no datatype SERIAL_PORT.
RTS_VALUE	BOOL	Valor a ser escrito no sinal RTS.
RTS_EN	BOOL	Habilita a escrita do parâmetro RTS_VALUE.
DTR_VALUE	BOOL	Valor a ser escrito no sinal DTR.
DTR_EN	BOOL	Habilita a escrita do parâmetro DTR_VALUE.
BREAK	BOOL	Caso seja verdadeiro, habilita lógica 0 (ocupado) na linha de transmissão.

Tabela 4-107. Parâmetros de Entrada Serial_SET_CTRL

Parâmetros de saída	Tipo	Descrição
DONE	BOOL	Essa variável é verdadeira quando o bloco funcional for executado por completo, caso contrário, ela é falsa.
EXEC	BOOL	Essa variável é verdadeira enquanto o bloco funcional estiver sendo executado, caso contrário, ela é falsa.
ERROR	BOOL	Essa variável é verdadeira quando o bloco funcional concluir a sua execução com algum erro, caso contrário, ela é falsa. Está vinculada à variável DONE, pois seu estado é exibido após a conclusão do bloco funcional.
STATUS	SERIAL_STATUS	Caso a variável ERROR seja verdadeira, a estrutura STATUS exibirá o erro encontrado na execução do bloco funcional. Os status, já descritos no datatype SERIAL_STATUS, possíveis são: - NO_ERROR - ILLEGAL_SERIAL_PORT - PORT_BUSY - HW_ERROR_UART - HW_ERROR_REMOTE - FB_SET_CTRL_NOT_ALLOWED - NOT_CONFIGURED

Tabela 4-108. Parâmetros de Saída Serial_SET_CTRL

Exemplo de utilização em Linguagem ST, após a biblioteca ser inserida no projeto e a porta serial ser configurada:

```
PROGRAM MainPrg
VAR

   Set_Control: SERIAL_SET_CTRL;
   Port: SERIAL_PORT := COM1;
   Status: SERIAL_STATUS;
END_VAR
```

```
//ENTRADAS:

Set_Control.REQUEST := TRUE;
Set_Control.PORT := Port;
Set_Control.RTS_VALUE := FALSE;
Set_Control.RTS_EN := FALSE;
Set_Control.DTR_VALUE := FALSE;
Set_Control.DTR_EN := FALSE;
Set_Control.BREAK := FALSE;
//FUNÇÃO:
Set_Control();
//SAÍDAS:
Set_Control.DONE;
Set_Control.EXEC;
Set_Control.ERROR;
Status := Set_Control.STATUS;          //Caso seja necessário tratar o
erro.
```

SERIAL_TX

Esse bloco funcional é utilizado para transmitir um buffer de dados pela porta serial, sendo que o mesmo somente é finalizado depois de todos os bytes serem transmitidos ou após o time-out (gera alguns erros).

Figura 4-78. Bloco para Transmitir Valores pela SERIAL_TX

Parâmetros de entrada	Tipo	Descrição
REQUEST	BOOL	Essa variável, quando verdadeira, habilita o uso do bloco funcional.
PORT	SERIAL_PORT	Seleciona a porta serial, conforme descrito no datatype SERIAL_PORT.
TX_BUFFER_POINTER	POINTER TO BYTE	Ponteiro de um array de bytes para transmitir os valores do buffer.
TX_BUFFER_LENGTH	UINT	Especifica o número de caracteres a serem transmitidos pelo array de bytes (tamanho máximo da fila TX é igual a 1024 caracteres).
TX_TIMEOUT	UINT	Especifica o time-out para completar a transmissão, incluindo a fase de handshake. O valor especificado, em ms, deve ser positivo e diferente de zero.
DELAY_BEFORE_TX	UINT	Especifica o atraso, em ms, entre a chamada do bloco funcional e o início da transmissão. Essa variável pode ser utilizada em comunicações com alguns modens.
CLEAR_RX_BEFORE_TX	BOOL	Quando verdadeiro, a fila RX e a UART FIFO RX são limpas antes de iniciar a transmissão. Esse comportamento é típico de protocolos mestre/escravo half-duplex.

Tabela 4-109. Parâmetros de Entrada SERIAL_TX

Parâmetros de saída	Tipo	Descrição
DONE	BOOL	Essa variável é verdadeira quando o bloco funcional for executado por completo, caso contrário, ela é falsa.
EXEC	BOOL	Essa variável é verdadeira enquanto o bloco funcional estiver sendo executado, caso contrário, ela é falsa.

ERROR	BOOL	Essa variável é verdadeira quando o bloco funcional concluir a sua execução com algum erro, caso contrário, ela é falsa. Está vinculada à variável DONE, pois seu estado é exibido após a conclusão do bloco.
STATUS	SERIAL_STATUS	Caso a variável ERROR seja verdadeira, a estrutura STATUS exibirá o erro encontrado na execução do bloco. Os status, já descritos no datatype SERIAL_STATUS, possíveis são: - NO_ERROR - ILLEGAL_SERIAL_PORT - PORT_BUSY - HW_ERROR_UART - HW_ERROR_REMOTE - ILLEGAL_TX_BUFF_LENGTH - ILLEGAL_TIMEOUT - CTS_TIMEOUT_ON - CTS_TIMEOUT_OFF - TX_TIMEOUT_ERROR - NOT_CONFIGURED.
TX_TRANSMITTED	UINT	Retorna o número de bytes transmitidos, o qual deve ser igual ao TX_BUFFER_LENGTH, mas pode ser menor caso ocorra algum erro durante a transmissão.

Tabela 4-110. Parâmetros de Saída SERIAL_TX

Exemplo de utilização em Linguagem ST, após a biblioteca ser inserida no projeto e a porta serial ser configurada:

```
PROGRAM MainPrg
VAR

   Transmit: SERIAL_TX;
   Port: SERIAL_PORT := COM1;
   Buffer_Pointer: ARRAY [0..9] OF BYTE := [0,1,2,3,4,5,6,7,8,9];
   Status: SERIAL_STATUS;
END_VAR

//ENTRADAS:
Transmit.REQUEST := TRUE;
Transmit.PORT := Port;
Transmit.TX_BUFFER_POINTER := ADR(Buffer_Pointer);
Transmit.TX_BUFFER_LENGTH := 10;
Transmit.TX_TIMEOUT := 10000;
Transmit.DELAY_BEFORE_TX := 1000;
Transmit.CLEAR_RX_BEFORE_TX := TRUE;
//FUNÇÃO:
Transmit();
//SAÍDAS:
Transmit.DONE;
Transmit.EXEC;
Transmit.ERROR;
Status := Transmit.STATUS;       //Caso seja necessário tratar o erro.
Transmit.TX_TRANSMITTED;
```

Atualização de Entradas e Saídas

Funcionalidade utilizada para atualizar entradas e saídas no decorrer do aplicativo, não sendo necessário aguardar até que seja completado um ciclo.

Quando os blocos funcionais para atualizar as entradas e saídas não são utilizados, a atualização é realizada a cada ciclo da MainTask.

ATENÇÃO:
Na inicialização de uma UCP da série Nexto, as entradas e saídas somente estarão atualizadas para leitura e preparadas para escrita quando a MainTask for executada. Todas as demais tarefas do sistema que executarem antes da MainTask estarão com as entradas e as saídas inválidas.

REFRESH_INPUT

Essa função é utilizada para atualizar as entradas do módulo especificado, sem aguardar o ciclo ser completado. É importante ressaltar que os filtros, configurados no MasterTool IEC XE, e o tempo de atualização das entradas do módulo, deverão ser considerados no tempo efetivo de atualização das entradas no aplicativo desenvolvido pelo usuário.

ATENÇÃO:
A função de REFRESH_INPUT deve ser utilizada somente na tarefa MainTask.
Para executar a atualização de entradas em outras tarefas, a opção *Habilita atualização de E/S por tarefa* deve ser selecionada, mais informações a respeito desta opção podem ser obtidas na Tabela 4-1.

ATENÇÃO:
A função de REFRESH_INPUT não suporta a atualização de entradas que tenham sido mapeadas para variáveis simbólicas. Para o correto funcionamento é necessário que a entrada esteja mapeada para uma variável dentro da memória de variáveis de entrada de representação direta (%I).

Figura 4-79. Função para Atualizar as Entradas

Parâmetros de entrada	Tipo	Descrição
byRackNumber	BYTE	Número do bastidor.
bySlotNumber	BYTE	Número da posição, na qual se encontra o módulo.

Tabela 4-111. Parâmetros de Entrada REFRESH_INPUT

Possíveis ERRORCODE:

- NoError: Execução com sucesso.
- IOModuleAbsent: O módulo está ausente.
- IOModuleNotConfigured: O módulo não foi configurado.
- ParameterMismatch: Esse erro é retornado caso a função seja chamada para um módulo que possua somente saídas, ou também caso a opção *Sempre atualizar variáveis* (localizada na tela de configuração do módulo, aba Mapeamento de E/S) não esteja marcada.
- InputReadFail: Falha crítica interna no módulo (o frame transmitido pela função não foi retornado dentro do time-out estipulado).
- FrameTransmitError: Falha crítica interna no módulo (erro na transmissão do frame da função).
- BusBusy: Falha crítica interna no módulo (o barramento não está habilitado para transmitir o frame da função).

Exemplo de utilização em Linguagem ST:

```
PROGRAM MainPrg
VAR

    Info: ERRORCODE;
```

```
byRackNumber: BYTE;
bySlotNumber: BYTE;
END_VAR
//ENTRADAS:
byRackNumber := 0;
bySlotNumber := 10;
//FUNÇÃO:
Info := REFRESH_INPUT (byRackNumber, bySlotNumber); //Chamada da
função.
//Variável 'Info' recebe possíveis erros da função.
```

REFRESH_OUTPUT

Essa função é utilizada para atualizar as saídas do módulo especificado, sem aguardar o ciclo ser completado. É importante ressaltar que o tempo de atualização das saídas do módulo deverá ser considerado no tempo efetivo de atualização das saídas no aplicativo desenvolvido pelo usuário.

ATENÇÃO: A função de REFRESH_OUTPUT deve ser utilizada somente na tarefa MainTask. Para executar a atualização de saídas em outras tarefas, a opção *Habilita atualização de E/S por tarefa* deve ser selecionada, mais informações a respeito desta opção podem ser obtidas na Tabela 4-1.

ATENÇÃO: A função de REFRESH_OUTPUT não suporta a atualização de saídas que tiverem sido mapeadas para variáveis simbólicas. Para o correto funcionamento é necessário que a saída esteja mapeada para uma variável dentro da memória de variáveis de saída de representação direta (%Q).

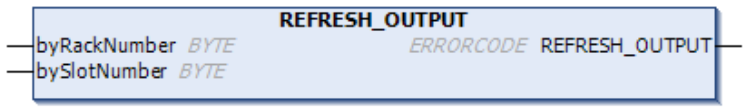

Figura 4-80. Função para Atualizar as Saídas

Parâmetros de entrada	Tipo	Descrição
byRackNumber	BYTE	Número do bastidor.
bySlotNumber	BYTE	Número da posição, na qual se encontra o módulo.

Tabela 4-112. Parâmetros de Entrada REFRESH_OUTPUT

Possíveis ERRORCODE:

- NoError: Execução com sucesso.
- IOModuleAbsent: O módulo foi configurado, porém está ausente.
- IOModuleNotConfigured: O módulo não foi configurado.
- ParameterMismatch: Esse erro é retornado caso a função seja chamada para um módulo que possua somente entradas, ou também caso a opção *Sempre atualizar variáveis* (localizada na tela de configuração do módulo, aba Mapeamento de E/S) não esteja marcada.
- OutputWriteFail: Falha crítica interna no módulo (o frame transmitido pela função não foi retornado dentro do time-out estipulado).
- FrameTransmitError: Falha crítica interna no módulo (erro na transmissão do frame da função).
- BusBusy: Falha crítica interna no módulo (o barramento não está habilitado para transmitir o frame da função).

Exemplo de utilização em Linguagem ST:

```
PROGRAM MainPrg
VAR

    Info: ERRORCODE;
    byRackNumber: BYTE;
    bySlotNumber: BYTE;
END_VAR
//ENTRADAS:
byRackNumber := 0;
bySlotNumber := 10;
//FUNÇÃO:
//Chamada da função.
Info := REFRESH_OUTPUT (byRackNumber, bySlotNumber);
//Variável 'Info' recebe possíveis erros da função.
```

Bloco Funcional PID

O bloco funcional PID é utilizado para controlar um processo real. O bloco está presente na biblioteca NextoPID, a qual deve ser adicionada ao projeto (para realizar o procedimento de inserção de uma biblioteca, consultar o Manual de Programação IEC 61131 – MP399048, capítulo Bibliotecas).

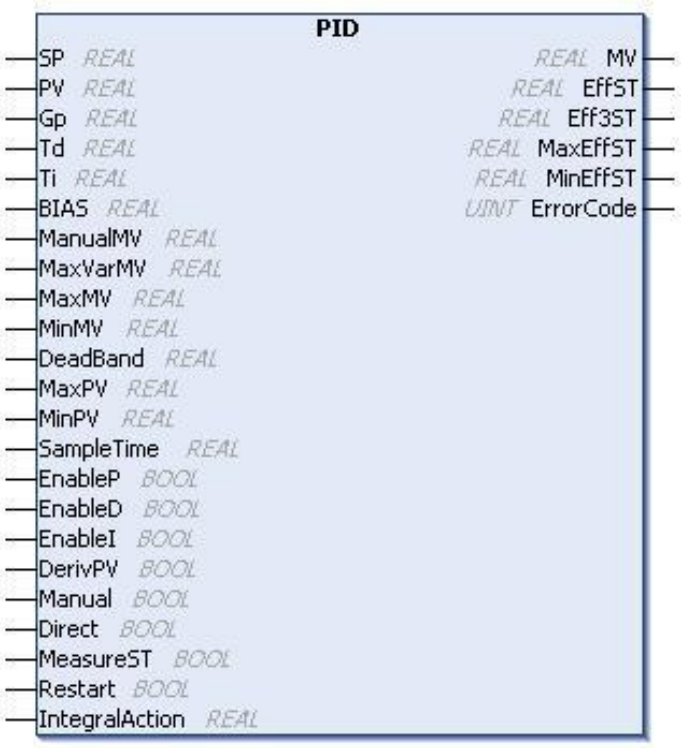

Figura 4-81. Bloco Funcional PID

Parâmetros de entrada	Tipo	Descrição
SP	REAL	Setpoint. A unidade e o intervalo devem ser os mesmos que o PV, pois as duas variáveis podem ser comparadas.
PV	REAL	Variável de processo. A unidade e o intervalo devem ser os mesmos que o SP, pois as duas variáveis podem ser comparadas.
Gp	REAL	Ganho proporcional utilizado para calcular a ação proporcional do bloco funcional PID.
Td	REAL	Tempo Derivativo, em segundos, utilizado para calcular a ação derivativa do bloco funcional PID.
Ti	REAL	Tempo Integral, em segundos, utilizado para calcular a ação integral do bloco funcional PID.

BIAS	REAL	Compensação adicionada à variável manipulada.
ManualMV	REAL	Valor atribuído à variável manipulada, quando utilizado o modo manual.
MaxVarMV	REAL	Máxima variação da variável manipulada, entre o ciclo atual e o ciclo anterior. Caso seja zero ou negativa, o bloco funcional PID não tem limite de variação de MV.
MaxMV	REAL	Máximo valor da variável manipulada. Caso o valor calculado seja maior do que o configurado, o MV será igual ao MaxMV.
MinMV	REAL	Mínimo valor da variável manipulada. Caso o valor calculado seja menor do que o configurado, o MV será igual ao MinMV.
DeadBand	REAL	Banda Morta. Mínimo valor de erro que irá causar a correção de MV em modo automático, ou seja, pequenos erros (menores que DeadBand) não causarão alterações na variável manipulada.
MaxPV	REAL	Máximo valor da variável de processo. Caso o valor de PV seja maior do que o configurado, o bloco funcional PID irá parar o cálculo e será gerado um código de erro na saída.
MinPV	REAL	Mínimo valor da variável de processo. Caso o valor de PV seja menor do que o configurado, o bloco funcional PID irá parar o cálculo e será gerado um código de erro na saída.
SampleTime	REAL	Tempo de amostragem. Define o período de chamada do bloco funcional PID, em segundos, podendo variar de 0,001 s a 1000 s. Esse parâmetro é desconsiderado se o MeasureST for verdadeiro.
EnableP	BOOL	Quando verdadeiro, habilita a ação proporcional do bloco funcional PID. Caso seja falso, a ação proporcional é zerada.
EnableD	BOOL	Quando verdadeiro, habilita ação derivativa do bloco funcional PID. Caso seja falso, a ação derivativa é zerada.
EnableI	BOOL	Quando verdadeiro, habilita a ação integral do bloco funcional PID. Caso seja falsa, a ação integral é zerada.
DerivPV	BOOL	Quando verdadeiro, a ação derivativa é calculada na variável de processo, sendo diferente de zero somente quando PV é alterado. Caso seja falso, a ação derivativa é calculada no erro, sendo dependente das variáveis SP e PV.
Manual	BOOL	Quando verdadeiro, habilita o modo manual. Caso seja falso, habilita o modo automático. O modo de controle do bloco funcional PID afeta a maneira como o MV e a ação integral são calculadas.
Direct	BOOL	Quando verdadeiro, seleciona-se o controle direto, fazendo com que MV seja somado na resposta para ser incluído no PV. Caso seja falso, seleciona-se o controle reverso, fazendo com que MV seja diminuído da resposta para ser incluído no PV.
MeasureST	BOOL	Quando verdadeiro, o tempo de amostragem é medido. Caso seja falso, o tempo de amostragem é informado pelo usuário na variável SampleTime.
Restart	BOOL	Quando verdadeiro, o bloco funcional PID é reiniciado, inicializando todas as variáveis. Também pode ser utilizado para limpar as ações integral e derivativa; e os códigos de erro na saída do bloco.
IntegralAction	REAL	Armazena a ação integral, a qual é eliminada em estado de erro.

Tabela 4-113. Parâmetros de Entrada PID

Parâmetros de saída	Tipo	Descrição
MV	REAL	Variável manipulada.

EffST	REAL	Tempo efetivo de amostragem, em segundos, utilizado para os cálculos de ação derivativa e taxa limite de MV.
Eff3ST	REAL	Tempo efetivo de amostragem dos três últimos ciclos, em segundos, utilizado para os cálculos de ação derivativa.
MaxEffST	REAL	Máximo valor do tempo efetivo de amostragem, em segundos, desde a inicialização do bloco funcional PID.
MinEffST	REAL	Mínimo valor do tempo efetivo de amostragem, em segundos, desde a inicialização do bloco funcional PID.
ErrorCode	UINT	Código de erro exibido pelo bloco funcional PID. Para removê-lo, basta solucionar o problema e reiniciar o bloco através da variável Restart. Abaixo, segue a descrição dos erros: 0: sem erro 1: MaxMV < MinMV 2: MaxPV < MinPV 3: PV > MaxPV 4: PV < MinPV 5: Ti < 0,001 s (com ação integral habilitada) 6: Td < 0 s (com ação derivativa habilitada) 7: Gp ≤ 0 8: MaxVarMV < 0 9: DeadBand < 0 10: SampleTime < 0,001 s ou SampleTime > 1000 s (com MeasureST = falso)

Tabela 4-114. Parâmetros de Saída PID

A Figura 4-82 mostra o diagrama de blocos de um laço PID, conforme a execução da UCP Nexto.

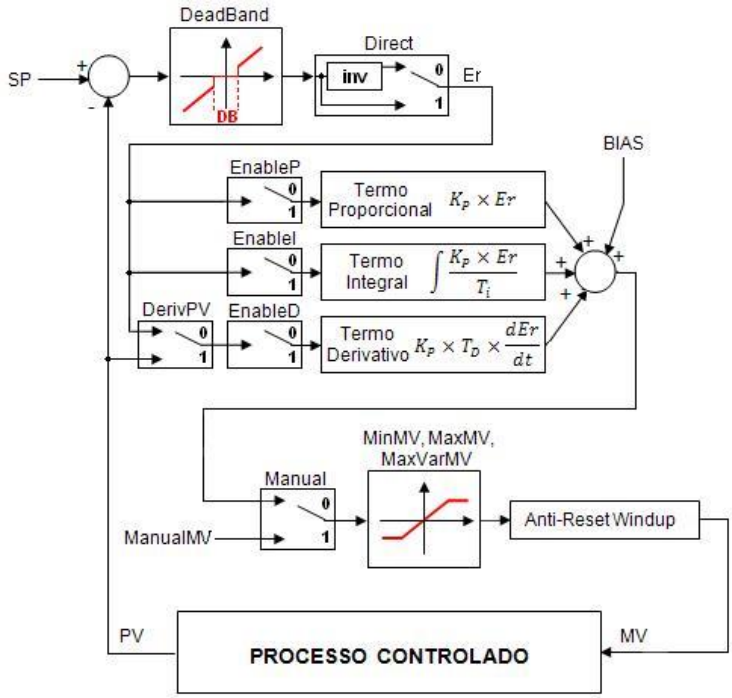

Figura 4-82. Diagrama PID

Timer Retentivo

O Timer Retentivo é um bloco funcional desenvolvido para aplicações como relógios de linha de produção, as quais necessitam armazenar o seu valor e reiniciar a contagem do mesmo ponto em caso de falha na alimentação. Os valores, guardados pelo bloco funcional somente serão zerados em caso de um Reset a Frio, Reset Origem ou o Download de uma nova aplicação (ver Manual de Utilização

do MasterTool IEC XE - MU299048), sendo que os contadores continuam em funcionamento mesmo que a aplicação esteja parada (Modo Stop).

ATENÇÃO: É importante destacar que, para o correto funcionamento dos blocos funcionais Timer Retentivo, as variáveis de controle devem ser declaradas como retentivas (VAR RETAIN). Também é importante ressaltar que em modo simulação os blocos funcionais Timer Retentivo não são executados adequadamente em virtude de necessitarem da UCP Nexto para o correto comportamento.

Abaixo, são descritos os três tipos de blocos disponíveis na biblioteca NextoStandard do software MasterTool IEC XE (para realizar o procedimento de inserção de uma biblioteca, consultar o Manual de Programação IEC 61131 – MP399048, capítulo Bibliotecas).

TOF_RET

O bloco funcional TOF_RET implementa um tempo de atraso para desabilitar uma saída. Quando a entrada IN tem seu estado alterado de verdadeiro (TRUE) para falso (FALSE), ou seja, uma borda de descida, o tempo especificado PT irá transcorrer até que a saída Q também seja falsa (FALSE). Quando a entrada IN tem nível lógico 1 (TRUE), a saída Q também permanecerá no mesmo estado (TRUE), mesmo que isso aconteça no meio de uma contagem. O tempo PT pode ser alterado durante a contagem, pois o bloco funcional assumirá o novo valor, desde que a contagem não tenha chegado ao final. A Figura 4-83 representa o bloco TOF_RET e a Figura 4-84 mostra o comportamento gráfico do mesmo.

Figura 4-83. Bloco Funcional TOF_RET

Parâmetros de entrada	Tipo	Descrição
IN	BOOL	Essa variável, quando recebe uma borda de descida, habilita a contagem do bloco funcional.
PT	TIME	Essa variável especifica o limite de contagem do bloco funcional (tempo de atraso).

Tabela 4-115. Parâmetros de Entrada TOF_RET

Parâmetros de saída	Tipo	Descrição
Q	BOOL	Essa variável vai para FALSE assim que a variável PT (tempo de atraso) atinge o seu valor máximo.
ET	TIME	Essa variável exibe o valor atual do tempo de atraso.

Tabela 4-116. Parâmetros de Saída TOF_RET

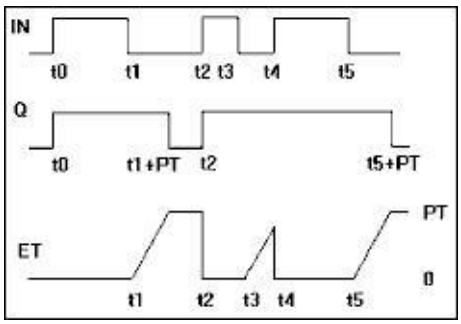

Figura 4-84. Comportamento Gráfico do Bloco Funcional TOF_RET

Exemplo de utilização em linguagem ST:

```
PROGRAM MainPrg
VAR RETAIN

   bStart : BOOL := TRUE;
   TOF_RET : TOF_RET;
END_VAR

// Quando bStart=FALSE inicia contagem
TOF_RET( IN := bStart,
PT := T#20S);

// Executa ações ao final da contagem
IF (TOF_RET.Q = FALSE) THEN

   bStart := TRUE;
END_IF
```

TON_RET

O bloco funcional TON_RET implementa um tempo de atraso para habilitar uma saída. Quando a entrada IN tem seu estado alterado de falso (FALSE) para verdadeiro (TRUE), ou seja, uma borda de subida, o tempo especificado PT irá transcorrer até que a saída Q também seja verdadeira (TRUE). Quando a entrada IN tem nível lógico 0 (FALSE), a saída Q também permanecerá no mesmo estado (FALSE), mesmo que isso aconteça no meio de uma contagem. O tempo PT pode ser alterado durante a contagem, pois o bloco funcional assumirá o novo valor, desde que a contagem não tenha chegado ao final. A Figura 4-85 representa o bloco TON_RET e a Figura 4-86 mostra o comportamento gráfico do mesmo.

Figura 4-85. Bloco Funcional TON_RET

Parâmetros de entrada	Tipo	Descrição
IN	BOOL	Essa variável, quando recebe uma borda de subida, habilita a contagem do bloco funcional.
PT	TIME	Essa variável especifica o limite de contagem do bloco funcional (tempo de atraso).

Tabela 4-117. Parâmetros de Entrada TON_RET

Parâmetros de saída	Tipo	Descrição
Q	BOOL	Essa variável vai para TRUE assim que a variável PT (tempo de atraso) atinge o seu valor máximo.
ET	TIME	Essa variável exibe o valor atual do tempo de atraso.

Tabela 4-118. Parâmetros de Saída TON_RET

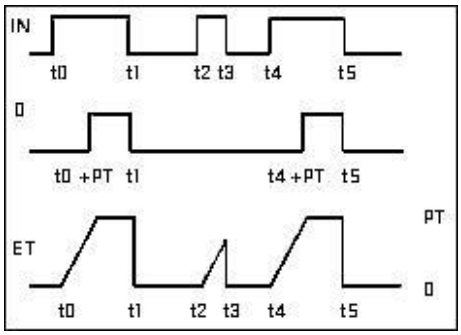

Figura 4-86. Comportamento Gráfico do Bloco Funcional TON_RET

Exemplo de utilização em linguagem ST:

```
PROGRAM MainPrg
VAR RETAIN
   bStart : BOOL;
   TON_RET : TON_RET;
END_VAR

// Quando bStart=TRUE inicia contagem
TON_RET( IN := bStart,
PT := T#20S);

// Executa ações ao final da contagem
IF (TON_RET.Q = TRUE) THEN
   bStart := FALSE;
END_IF
```

TP_RET

O bloco funcional TP_RET trabalha como um trigger. O timer, que inicia quando a entrada IN tem seu estado alterado de falso (FALSE) para verdadeiro (TRUE), ou seja, uma borda de subida; é incrementado até que o limite de tempo PT seja atingido. Durante a contagem, a saída Q é verdadeira (TRUE), caso contrário ela é falsa (FALSE). O tempo PT pode ser alterado durante a contagem, pois o bloco assumirá o novo valor, desde que a contagem não tenha chegado ao final. A Figura 4-87 representa o bloco TP_RET e a Figura 4-88 mostra o comportamento gráfico do mesmo.

Figura 4-87. Bloco Funcional TP_RET

Parâmetros de entrada	Tipo	Descrição
IN	BOOL	Essa variável, quando recebe uma borda de subida, habilita a contagem do bloco funcional.
PT	TIME	Essa variável especifica o limite de contagem do bloco funcional (tempo de atraso).

Tabela 4-119. Parâmetros de Entrada TP_RET

Parâmetros de saída	Tipo	Descrição
Q	BOOL	Essa variável é verdadeira durante a contagem. Caso contrário é falsa.
ET	TIME	Essa variável exibe o valor atual do tempo de atraso.

Tabela 4-120. Parâmetros de Saída TP_RET

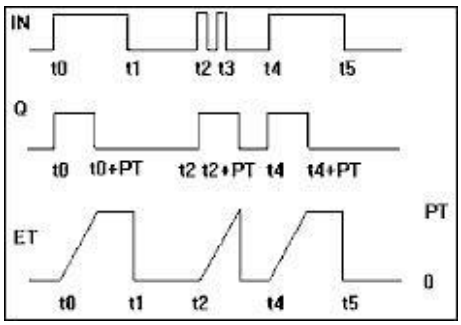

Figura 4-88. Comportamento Gráfico do Bloco Funcional TP_RET

Exemplo de utilização em linguagem ST:

```
PROGRAM MainPrg
VAR RETAIN
   bStart : BOOL;
   TP_RET : TP_RET;
END_VAR

// Configura TP_NR
TP_RET( IN := bStart,
PT := T#20S);

bStart := FALSE;

// Ações durante a contagem
IF (TP_RET.Q = TRUE) THEN
   // Executa enquanto o contador estiver ativado
ELSE
   // Executa somente quando o contador estiver desativado
END_IF
```

Timer Não-Redundante

O timer não redundante é utilizado em aplicações para a UCP NX3030 redundante que necessitam de um timer no programa não redundante de um half-cluster. Este timer não utiliza o timer IEC, portanto, não irá ser sincronizado no caso de o half-cluster reserva assumir o estado ativo e o ativo passar para reserva.

Abaixo, são descritos os três tipos de blocos já disponíveis na biblioteca NextoStandard do software MasterTool IEC XE (para realizar o procedimento de inserção de uma biblioteca, consultar o Manual de Programação IEC 61131 – MP399048, capítulo Bibliotecas).

TOF_NR

O bloco funcional TOF_NR implementa um tempo de atraso para desabilitar uma saída e tem o funcionamento e configuração parecidos com o bloco funcional TOF_RET, se diferenciando apenas por não ser redundante e nem retentivo.

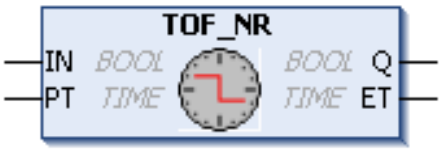

Figura 4-89. Bloco Funcional TOF_NR

Exemplo de utilização em linguagem ST:

```
PROGRAM NonSkippedProg
VAR
   bStart : BOOL := TRUE;
   TOF_NR : TOF_NR;
END_VAR

// Quando bStart=FALSE inicia contagem
TOF_NR( IN := bStart,
PT := T#20S);

// Executa ações ao final da contagem
IF (TOF_NR.Q = FALSE) THEN
   bStart := TRUE;
END_IF
```

TON_NR

O bloco funcional TON_NR implementa um tempo de atraso para habilitar uma saída e tem o funcionamento e configuração parecidos com o bloco funcional TON_RET, se diferenciando apenas por não ser redundante e nem retentivo.

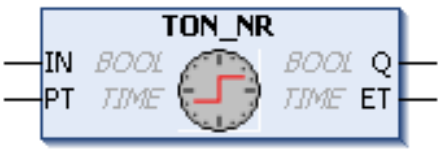

Figura 4-90. Bloco Funcional TON_NR

Exemplo de utilização em linguagem ST:

```
PROGRAM NonSkippedProg
VAR
   bStart : BOOL;
   TON_NR : TON_NR;
END_VAR

// Quando bStart=TRUE inicia contagem
TON_NR( IN := bStart,
PT := T#20S);
```

```
// Executa ações ao final da contagem
IF (TON_NR.Q = TRUE) THEN
   bStart := FALSE;
END_IF
```

TP_NR

O bloco funcional TP_NR trabalha como um trigger e tem o funcionamento e configuração parecidos com o bloco funcional TP_RET, se diferenciando apenas por não ser redundante e nem retentivo.

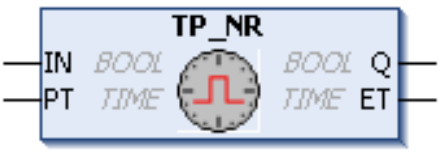

Figura 4-91. Bloco Funcional TP_NR

Exemplo de utilização em linguagem ST:

```
PROGRAM NonSkippedProg
VAR
   bStart : BOOL;
   TP_NR : TP_NR;
END_VAR

// Configura TP_NR
TP_NR( IN := bStart,
PT := T#20S);

bStart := FALSE;

// Ações durante a contagem
IF (TP_NR.Q = TRUE) THEN
   // Executa enquanto o contador estiver ativado
ELSE
   // Executa somente quando o contador estiver desativado
END_IF
```

Log de Usuário

Recurso que permite ao usuário criar seus próprios registros e gravar em arquivos de log no cartão de memória presente na UCP. Os arquivos são gerados em um diretório específico do cartão de memória no formato CSV, permitindo a visualização em editores de texto e planilhas. O separador adotado foi o caractere ponto e vírgula. Para mais informações sobre a utilização do cartão de memória, ver capítulo Configuração - Cartão de Memória.

Há duas funções disponíveis, uma para registrar informações e outra para remover todos os registros. A seguir uma descrição dos tipos de dados utilizados pelas funções:

Tipo de dado	Opção	Descrição
USER_LOG_EVENT_TYPES	USER_LOG_EVENT_ERROR	O usuário é livre para usar a melhor indicação conforme a severidade de sua mensagem de log.
	USER_LOG_EVENT_DEBUG	
	USER_LOG_EVENT_INFO	
	USER_LOG_EVENT_WARN	
USER_LOG_MESSAGE		Mensagem de log com o máximo de 150 caracteres.
USER_LOG_ERROR_CODES	USER_LOG_OK	A operação foi executada com sucesso.

	USER_LOG_FAILED	A operação não foi executada com sucesso. O motivo da falha pode ser verificado nos logs de sistema – ver capítulo Manutenção - Log de Sistema.
	USER_LOG_BUFFER_FULL	Mensagens estão sendo adicionadas além da capacidade de processamento.
	USER_LOG_NO_MEMORY	No momento, não houve recursos para executar a operação.
	USER_LOG_FILE_SYSTEM_ERROR	Houve algum erro no acesso ao cartão de memória ou não há espaço disponível. Informações do erro pode ser verificado nos logs de sistema – ver capítulo Manutenção - Log de Sistema.
	USER_LOG_NO_MEMORY_CARD	Não há um cartão de memória presente na UCP.
	USER_LOG_MEMORY_CARD_FULL	Não há espaço livre disponível no cartão de memória.
	USER_LOG_PROCESSING	O recurso está ocupado executando a última operação, por exemplo, deletando todos os arquivos de log.

Tabela 4-121. Tipos de dados para Log de Usuário

A seguir, são descritas as duas funções disponíveis na biblioteca LibLogs no MasterTool IEC XE 1.40. Para realizar o procedimento de inserção de uma biblioteca, consultar o Manual de Programação IEC 61131 – MP399048, capítulo Bibliotecas.

ATENÇÃO:
Os Logs de Usuário estão disponíveis apenas a partir da versão 1.3.0.20 das UCPs da Série Nexto. Da mesma forma para utilizar esta funcionalidade é necessária à versão 1.40 ou superior do MasterTool IEC XE.

UserLogAdd

Essa função é utilizada para adicionar uma nova mensagem de log do usuário, acrescentando em uma nova linha ao arquivo de log no cartão de memória. A mensagem deve ter um tamanho máximo de 150 caracteres e o tipo de evento da mensagem. Variáveis da aplicação podem ser registradas utilizando conversão para string e concatenação com a mensagem principal. É adicionado automaticamente na mensagem a informação de data e hora em UTC (timestamp) com resolução de milissegundos em que o evento foi registrado. A informação de data e hora também é utilizada na formação dos nomes dos arquivos de log.

A função UserLogAdd pode ser utilizada para inserir várias mensagens dentro de uma mesma tarefa e também em tarefas diferentes da aplicação. Porém independente de cada execução da função na aplicação, sendo ela na mesma tarefa ou em tarefas diferentes, todas utilizam o mesmo recurso para gravar as mensagens desejadas. Por esse motivo é recomendado que a adição de mensagens utilizando a função UserLogAdd na aplicação seja realizada a cada 50 ms para evitar retorno de buffer cheio. Caso a função seja executada em períodos menores que o indicado, mas respeite o tempo médio de 50 ms entre cada adição de mensagem ao término do intervalo da tarefa, também evita o retorno de buffer cheio. Para que os logs sejam adicionados corretamente, é importante respeitar os limites de tempo quando o cartão é inserido e na inicialização da UCP, mencionados no capítulo Configuração - Cartão de Memória. Após a operação a função retorna as opções para o tipo de dado USER_LOG_ERROR_CODES, conforme a Tabela 4-121. Quando o retorno da função for diferente de USER_LOG_OK, a mensagem não foi registrada, devendo ser executada novamente a função UserLogAdd com a mensagem desejada. Em caso de retorno de falhas consecutivas de escrita, o cartão de memória pode estar danificado. A substituição por um cartão de memória íntegro garante que as últimas mensagens registradas serão gravadas no cartão que não está danificado, desde que a UCP não seja reiniciada.

A Figura 4-92 representa a função UserLogAdd e a Tabela 4-122 os parâmetros de entrada:

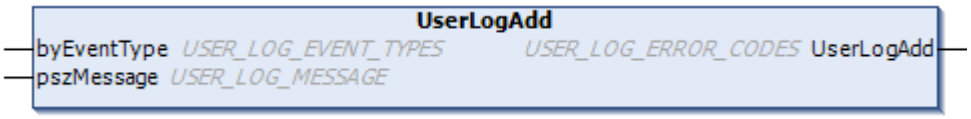

Figura 4-92. Função UserLogAdd

Parâmetros de entrada	Tipo	Descrição
byEventType	BYTE	Essa variável especifica o tipo de evento do log que está sendo adicionado conforme as opções para o tipo de dado USER_LOG_EVENT_TYPES.
pszMessage	USER_LOG_MESSAGE	Essa variável deve conter o conjunto de caracteres que compõe a mensagem a ser adicionada no arquivo de log. A mensagem deve conter no máximo 150 caracteres.

Tabela 4-122. Parâmetros de Entrada

Os arquivos de log são gerados e organizados no cartão de memória em um caminho de diretórios específico que depende do número de série da UCP e da versão de firmware instalada. Por exemplo, para uma UCP com número de série 445627 e versão de firmware 1.4.0.4, o local onde os arquivos de log devem ser gravados no cartão de memória é MemoryCard/UserLog/445627/1.4.0.4/.

Os nomes dos arquivos de log são formados pela data e hora (timestamp) da primeira mensagem. Exceto quando há algum problema para utilizar esse nome, como por exemplo, outro arquivo existente com o mesmo nome, nessa situação é utilizado à data e hora instantânea. O nome do arquivo segue o seguinte padrão: ano/mês/dia/hora/minuto/segundo/milissegundo.CSV. Caso um arquivo apresente problema de acesso por setor defeituoso e não seja possível continuar com escrita, será adicionado ao nome desse arquivo à extensão “corrupted” e um novo arquivo será criado. A quantidade de logs por arquivo não é fixa, variando em função do tamanho das mensagens. A quantidade de arquivos criados é limitada em 1024 com tamanho máximo de 1MB cada, portanto o cartão de memória necessita de 1GB de espaço livre. Quando atingir o limite 1024 arquivos criados no cartão de memória, durante a operação da UCP, os arquivos mais antigos são removidos para que os arquivos com logs mais recentes sejam preservados, mesmo nos casos de remoção manual parcial dos arquivos no diretório onde os arquivos estão sendo gravados.

A visualização dos arquivos de log pode ser realizada através de planilhas ou editores de texto convencionais. As informações concatenadas, para melhor visualização, podem utilizar ponto e vírgula entre as strings da mensagem para separá-las. Deve-se ter o cuidado na formatação de células com valores em ponto flutuante.

Exemplo de utilização em Linguagem ST:

```
PROGRAM MainPrg
VAR
   eLogError : USER_LOG_ERROR_CODES;
   sMessage :USER_LOG_MESSAGE;
END_VAR

IF (m_rTemperature > MAX_TEMPERATURE_ACCEPT) THEN
   sMessage := 'Temperatura acima do esperado: ';
   sMessage := concat(sMessage, REAL_TO_STRING(m_rTemperature));
   sMessage := concat(sMessage, '°');
   eLogError := UserLogAdd(USER_LOG_EVENT_WARN, sMessage);
//Variável ‘eLogError’ recebe possíveis erros da função.
END_IF
```

Exemplo de conteúdo do arquivo de log: (UserLog-201308271506245738.csv)

```
Model; NX3030
Serial number; 445627
```

```
Firmware version; 1.4.0.4

27/08/2013 15:06:24.5738; WARN; Temperatura acima do esperado: 25°
27/08/2013 16:37:45.3476; WARN; Temperatura acima do esperado: 25°
28/08/2013 09:10:55.4201; WARN; Temperatura acima do esperado: 26°
```

UserLogDeleteAll

A função UserLogDeleteAll realiza a exclusão dos arquivos de log presentes no diretório criado especificamente para a UCP em que está inserido o cartão de memória, ou seja, são excluídos apenas os logs contidos no diretório nomeado com a versão de firmware da UCP que existe dentro do diretório com a versão de série da UCP. Os arquivos de log excluídos são apenas os arquivos existentes no momento da montagem do cartão de memória e os gerados pela função UserLogAdd. Registros de outras UCPs e arquivos adicionados manualmente pelo usuário durante a execução não são apagados. A Figura 4-93 representa a função UserLogDeleteAll.

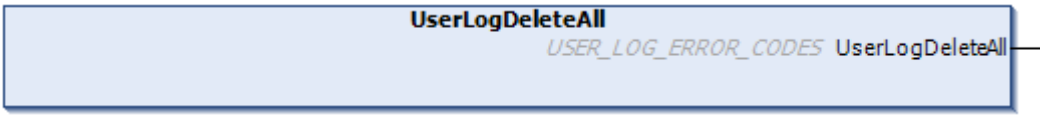

Figura 4-93. Função UserLogDeleteAll

Exemplo de utilização em Linguagem ST:

```
PROGRAM MainPrg
VAR

   eLogError : USER_LOG_ERROR_CODES;
END_VAR

IF (m_DeleteLogs = TRUE) THEN

   eLogError := UserLogDeleteAll();m_DeleteLogs := FALSE;
//Variável 'eLogError' recebe possíveis erros da função.
END_IF
```

ATENÇÃO: O retorno da função UserLogDeleteAll não indica operação concluída, apenas confirmação de execução que pode levar um período de tempo grande caso existam centenas de arquivos de log no diretório. A função para registro de novos logs de usuário estará indisponível nesse momento, retornando a opção USER_LOG_PROCESSING para qualquer operação. O resultado da operação também pode ser verificado no log de sistema.

SNMP

Introdução

SNMP (Simple Network Management Protocol) é um protocolo muito utilizado por administradores de redes por fornecer importantes informações e diagnósticos de equipamentos presentes em determinada rede Ethernet.

Este protocolo usa o conceito de agente e gerente, no qual o gerente envia requisições de leitura ou escrita de determinados objetos para o agente. Através de uma MIB (Management Information Base) o gerente tem conhecimento dos objetos existentes no agente, e dessa forma poderá fazer requisições desses objetos, respeitando as permissões de leitura ou escrita dos mesmos.

MIB é uma coleção de informações organizadas hierarquicamente, onde cada objeto desta árvore é chamado de OID (Object Identifier).

Para todos os equipamentos com SNMP, é obrigatório o suporte a MIB-II. Nesta MIB estão descritas as principais informações para gerenciamento de redes Ethernet.

SNMP nas UCPs Nexto

As UCPs da Série Nexto se comportam como agentes na comunicação SNMP. As informações disponibilizadas através do SNMP não podem ser manipuladas ou acessadas através da aplicação do usuário, sendo necessário um gerente SNMP externo para efetuar o acesso. A Tabela 4-123 contém os objetos disponíveis nas UCPs Nexto. Esta funcionalidade está disponível a partir da versão de firmware 1.4.0.33, das UCPs da Série Nexto, possuindo suporte aos protocolos SNMPv1, SNMPv2c e SNMPv3, além de suporte a MIB-II, onde objetos obrigatórios são descritos na RFC-1213.

OID	Nome	Descrição
1.3.6.1.2.1.1	System	Contêm nome, descrição, localização e outras informações de identificação do equipamento.
1.3.6.1.2.1.2	Interfaces	Contêm informações das interfaces de rede do equipamento. A tabela ifTable (OID 1.3.6.1.2.1.2.2) possui os indexes 6 e 7 disponíveis, pelos quais podem ser visualizadas as estatísticas das interfaces de rede NET 1 e NET 2, respectivamente, das UCPs da série Nexto.
1.3.6.1.2.1.3	At	Contêm informações das últimas conexões requisitadas ao agente.
1.3.6.1.2.1.4	Ip	Contêm estatísticas de conexões utilizando protocolo IP.
1.3.6.1.2.1.5	Icmp	Contêm estatísticas para o protocolo ICMP.
1.3.6.1.2.1.6	Tcp	Contêm estatísticas para o protocolo TCP.
1.3.6.1.2.1.7	Udp	Contêm estatísticas para o protocolo UDP.
1.3.6.1.2.1.11	Snmp	Contêm estatísticas para o protocolo SNMP.
1.3.6.1.2.1.31	ifMib	Extensão para Interfaces, OID 1.3.6.1.2.1.2.

Tabela 4-123. Objetos da MIB-II - Agente SNMP da Série Nexto

Por padrão, o agente SNMP está ativado, ou seja, o serviço é inicializado no momento em que a UCP é iniciada. O acesso às informações do agente se dá através das interfaces Ethernet NET 1 e NET 2 das UCPs da Série Nexto na porta TCP 161. Portanto quando o serviço estiver ativo, as informações do agente poderão ser acessadas através de qualquer uma das duas interfaces Ethernet, conforme a disponibilidade da UCP em uso. Não é possível disponibilizar informações do agente através de interfaces Ethernet dos módulos NX5000. Na Figura 4-94 é mostrado um exemplo de gerente SNMP, no qual são lidos alguns valores.

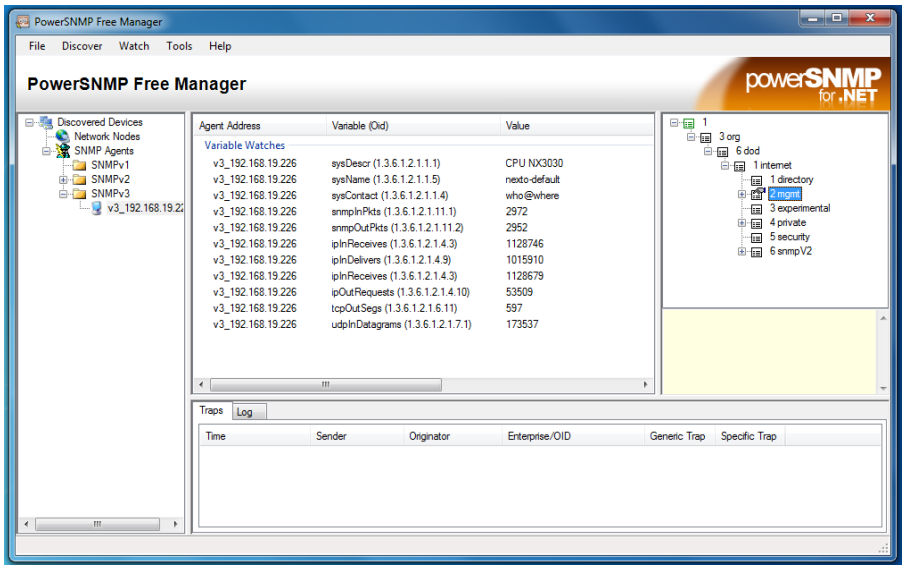

Figura 4-94. Exemplo de Gerente SNMP

Para SNMPv3, na qual existe autenticação de usuário e senha para requisições via protocolo SNMP, é fornecido um usuário padrão descrito na seção Usuário e Comunidades SNMP.

Caso o usuário deseje desativar o serviço, alterar o usuário SNMPv3 ou as comunidades para SNMPv1/v2c pré-definidos, será necessário acessar a página web da UCP. Para mais detalhes consulte a seção Configuração.

Private MIB

Além de suporte a MIB-II, as UCPs da série Nexto oferecem suporte a Private MIB a partir da versão de firmware 1.4.0.25. Para isso foi reservada uma PEN (Private Enterprise Number) com o número 43427 exclusivo para produtos Altus S.A. Assim, todos os objetos privados da Série Nexto podem ser acessados através do OID .1.3.6.1.4.1.43427.1 (iso.org.dod.internet.private.enterprise.AltusSA.Nexto). Neste OID há três subdivisões pertinentes as UCPs Nexto, como pode ser visto na Figura 4-95.

Todos os objetos privados estão descritos nas MIBs ALTUS-NEXTO-NX3004-MIB, ALTUS-NEXTO-NX3010-MIB, ALTUS-NEXTO-NX3020-MIB e ALTUS-NEXTO-NX3030-MIB e podem ser encontradas em www.altus.com.br.

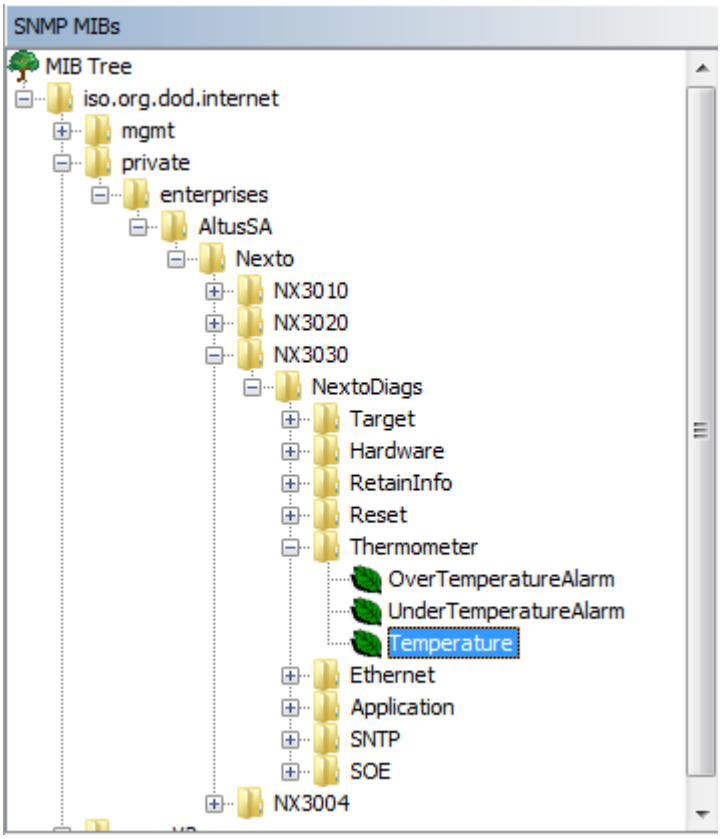

Figura 4-95. OID TreeView

Os objetos disponíveis via SNMP nas UCPs da série Nexto são diagnósticos já existentes, e que possuem importância para o gerenciamento de redes. Estes objetos podem ser visualizados na Tabela 4-124.

Para acessá-los através de um gerente SNMP o usuário deverá fazer requisições a partir do OID .1.3.6.1.4.1.43427.1.4.1 para NX3004, .1.3.6.1.4.1.43427.1.1.1 para NX3010, .1.3.6.1.4.1.43427.1.2.1 para NX3020 e .1.3.6.1.4.1.43427.1.3.1 para NX3030.

Por exemplo, se o usuário desejar monitorar a temperatura interna de uma UCP NX3030, o OID correspondente nesse caso será .1.3.6.1.4.1.43427.1.3.1.5.3 (iso.org.dod.internet.private.enterprise.AltusSA.Nexto.NX3030.NextoDiags.Thermometer.Temperature).

Grupo	Diagnóstico	Descrição
Target	CPUModel	NX30xx.
	CPUVersion	Versão do firmware.
	BootloaderVersion	Versão do bootloader.
	AuxprocVersion[1]	Versão do processador auxiliar.
Hardware	AuxprocFailure[1]	Falha na comunicação entre o processador auxiliar e o processador principal.
	RTCFailure	O processador principal não está habilitado para comunicar com o RTC (relógio da UCP).
	ThermometerFailure	Falha na comunicação entre o termômetro e o processador principal.
	LCDFailure[1]	Falha na comunicação entre o visor LCD e o processador principal.
RetainInfo	CPUInitStatus	Status da Inicialização da UCP:

			01: Hot start 02: Warm Start 03: Cold Start Obs.: Essas variáveis são reinicializadas em todas as energizações.
		CPUColdStartCounter	Contador de Inicializações a frio: Será incrementado somente devido à retirada a quente da UCP no barramento e não devido ao comando de Reset a Frio do MasterTool IEC XE (0 a 65535).
		CPUWarmStartCounter	Contador de Inicializações a quente: Será incrementado somente durante a sequência de energizações do sistema e não devido ao comando de Reset a Quente do MasterTool IEC XE (0 a 65535).
		CPUHotStartCounter[1]	Contador de distúrbios menores do que o tempo de suporte à falhas na alimentação da UCP (0 a 65535).
		RTSResetCounter	Contador de reset efetuados pelo RTS (Runtime System) (0 a 65535).
Reset		BrownOut	A UCP foi reiniciada devido a uma falha na alimentação na última inicialização.
		Watchdog	A UCP foi reiniciada devido ao cão-de-guarda ativo na última inicialização.
Thermometer		UnderTemperatureAlarm	Alarme gerado devido à temperatura interna estar em 0 ºC ou abaixo de 0 ºC.
		OverTemperatureAlarm	Alarme gerado devido à temperatura interna estar em 85 ºC ou acima de 85 ºC.
		Temperature	Temperatura interna.
Ethernet	NET 1	ModbusRtuEthClient1	Cliente MODBUS RTU via TCP.
		ModbusEthClient1	Cliente MODBUS TCP.
		ModbusRtuEthServer1	Servidor MODBUS RTU via TCP.
		ModbusEthServer1	Servidor MODBUS TCP.
	NET 2	ModbusRtuEthClient2[1]	Cliente MODBUS RTU via TCP.
		ModbusEthClient2[1]	Cliente MODBUS TCP.
		ModbusRtuEthServer2[1]	Servidor MODBUS RTU via TCP.
		ModbusEthServer2[1]	Servidor MODBUS TCP.
Application		CPUState	Informa o estado de operação da UCP: 00: Aplicação parada (Modo Stop) 01: Aplicação em execução (Modo Run) 02: Aplicação parada em Breakpoint 03: Aplicação parada por Exceção
		ForcedIOs	Existem um ou mais pontos de IO forçados.
SNTP		ServiceEnabled	Serviço SNTP habilitado.
		ActiveTimeServer	Indica qual servidor está ativo: 00: Nenhum servidor ativo 01: Servidor primário ativo 02: Servidor secundário ativo

		PrimaryServerDownCount	Contador de vezes que o servidor primário esteve indisponível (0 a 65535).
		SecondaryServerDownCount	Contador de vezes que o servidor secundário esteve indisponível (0 a 65535).
		RTCTimeUpdatedCount	Contador de vezes que o RTC foi atualizado pelo serviço SNTP (0 a 4294967295).
		LastUpdateSuccessfull	Indica status da última atualização: 00: Não foi atualizado 01: Última atualização falhou 02: Última atualização teve sucesso
		LastUpdateTimeServer	Indica qual servidor foi utilizado na última atualização: 00: Nenhuma atualização 01: Servidor primário 02: Servidor secundário
		LastUpdateTimeDayOfMonth	Dia da última atualização do RTC.
		LastUpdateTimeMonth	Mês da última atualização do RTC.
		LastUpdateTimeYear	Ano da última atualização do RTC.
		LastUpdateTimeHours	Hora da última atualização do RTC.
		LastUpdateTimeMinutes	Minuto da última atualização do RTC.
		LastUpdateTimeSeconds	Segundo da última atualização do RTC.
		LastUpdateTimeMilliseconds	Milissegundo da última atualização do RTC.
SOE	SOE1	ConnectionStatus1[1]	Status de conexão do cliente 01.
		OverflowStatus1[1]	Status da fila de eventos do cliente 01 FALSE - Não houve estouro TRUE - Limite da fila excedido
		EventsCounter1[1]	Contador de eventos na fila do cliente 01.
	SOE2	ConnectionStatus2[1]	Status de conexão do cliente 02.
		OverflowStatus2[1]	Status da fila de eventos do cliente 02 FALSE - Não houve estouro TRUE - Limite da fila excedido
		EventsCounter2[1]	Contador de eventos na fila do cliente 02.

Tabela 4-124. Diagnósticos via SNMP

(1) Estes diagnósticos não estão disponíveis na UCP NX3004.

ATENÇÃO: Os itens Ethernet NET2 e SOE estão disponíveis apenas para as UCPs NX3020 e NX3030.

Configuração

As configurações do SNMP podem ser alteradas através da página web, na aba Gerenciamento da UCP, no menu SNMP. Para ter acesso as configurações, o usuário deverá primeiro efetuar o login, conforme Figura 4-96.

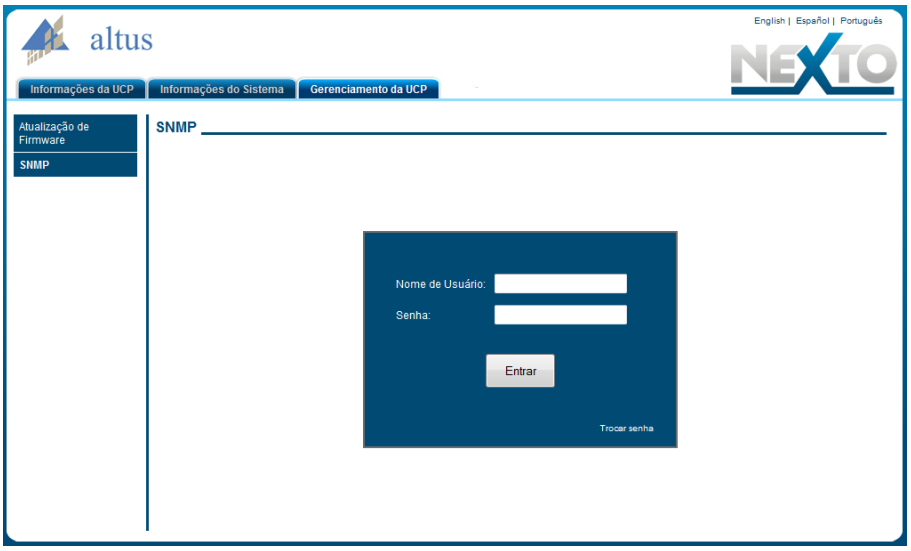

Figura 4-96. Tela de Login SNMP

Efetuado o login com sucesso, poderá ser visualizado o estado atual do serviço (ativado ou desativado), assim como as informações de usuário SNMPv3 e comunidades para SNMPv1/v2c.

O usuário poderá ativar ou desativar o serviço através de um checkbox na parte superior da tela.

É possível também alterar as informações de SNMPv3, clicando no botão Alterar logo abaixo das informações de usuário. Abrirá um formulário onde é necessário preencher o usuário e senha antigos, e o novo usuário e senha. As demais informações de usuário SNMPv3 não podem ser alteradas.

Para alterar os dados das comunidades SNMPv1/v2c, o processo é parecido, basta clicar no botão Alterar abaixo das informações das comunidades. Uma nova tela será aberta onde serão inseridos os novos dados para os campos rocommunity e rwcommunity. Caso o usuário deixe qualquer um dos campos em branco, a respectiva comunidade será desativada. Dessa forma, se o usuário deixar os 2 campos em branco, o acesso ao agente SNMP será possível somente através do SNMPv3.

Caso o usuário deseje retornar para as configurações padrão, será necessário reconfigurar manualmente as mesmas de acordo com a seção Usuário e Comunidades SNMP. Portanto, todas configuração SNMP atuais serão mantidas no processo de atualização de firmware. Estas opções podem ser visualizas na Figura 4-97.

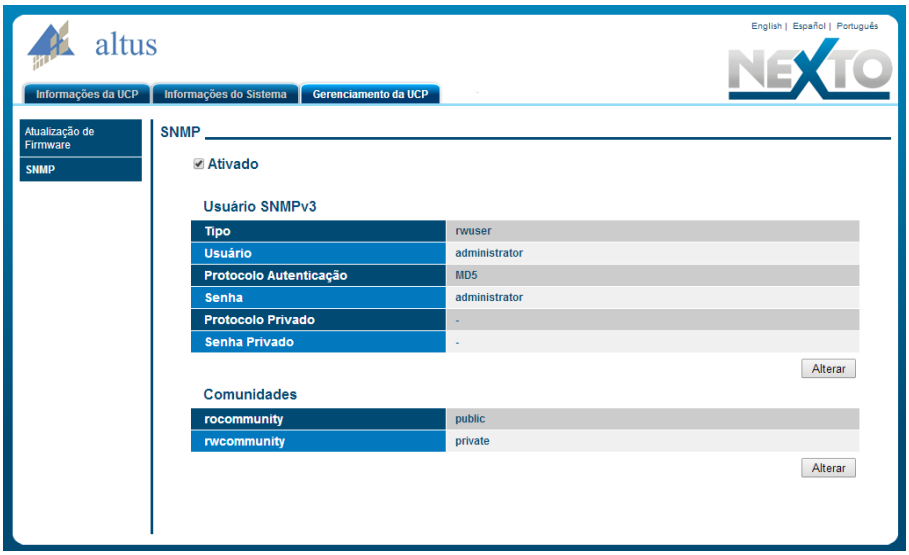

Figura 4-97. Tela de Configuração e Status SNMP

ATENÇÃO:
Caso as telas apresentadas forem diferentes das visualizadas no navegador, será necessário uma limpeza de cache do navegador.

ATENÇÃO:
O usuário e senha para login na página web das Configurações SNMP e para acessar o agente via protocolo SNMP são iguais.

Usuário e Comunidades SNMP

Para acesso ao SNMPv1/v2c das UCPs da Série Nexto, existem duas comunidades, conforme Tabela 4-125.

Comunidades	String Padrão	Tipo
rocommunity	Public	Somente leitura
rwcommunity	Private	Leitura e escrita

Tabela 4-125. Informações de Comunidades Padrão SNMPv1/v2c

É possível efetuar acesso ao SNMPv3 através do usuário padrão conforme Tabela 4-126.

Usuário	Tipo	Protocolo de Autenticação	Senha Autenticação	Protocolo Privado	Senha Privado
administrator	rwuser	MD5	Administrator	-	-

Tabela 4-126. Informações de Usuário Padrão SNMPv3

Para todas as configurações de comunidades, usuário e senha existem limites que devem ser observados na Tabela 4-127.

Item configurável	Tamanho mínimo	Tamanho máximo	Caracteres permitidos
rocommunity	-	30	[0-9][a-z][A-Z]@$*_.
rwcommunity	-	30	[0-9][a-z][A-Z]@$*_.
Usuário v3	-	30	[0-9][a-z][A-Z]@$*_.
Senha v3	8	30	[0-9][a-z][A-Z]@$*_.

Tabela 4-127. Limites para Configurações SNMP

Gerenciamento de Usuários e Direitos de Acesso

Fornece funções para definir contas dos usuários e configurar os direitos de acesso ao projeto e a UCP. Utilizando o software MasterTool IEC XE é possível criar e gerenciar usuários e grupos, configurando diferentes níveis de direitos de acesso ao projeto para estes.

Simultaneamente, as UCPs Nexto possuem um sistema de gerenciamento de permissões de usuário, que bloqueia ou permite certas ações para cada grupo de usuários na UCP. Para mais informações é necessário consultar o Manual de Utilização MasterTool IEC XE MT8500 – MU299048, capítulo Gerenciamento de Usuários e Direitos de Acesso.

5. Programação Inicial

O objetivo deste capítulo é auxiliar na programação e configuração das UCPs da Série Nexto, para que o usuário de os primeiros passos antes de iniciar a programação de um CP.

As UCPs da Série Nexto utilizam a norma padrão de linguagens IEC 61131-3, que são IL, ST, LD, SFC e FBD, e, além disso, uma linguagem extra, CFC. Pode-se separar essas linguagens em textuais e gráficas. O IL e ST são linguagens textuais e são semelhantes ao Assembly e a linguagem C. O LD, SFC, FBD e CFC são linguagens gráficas. LD usa a representação de blocos de relés e é similar ao diagrama de relés. SFC usa a representação de diagrama de sequência, possibilitando uma maneira fácil de enxergar as sequencias de eventos. FBD e o CFC usam um conjunto de blocos funcionais, permitindo uma visão clara das funções exercidas por cada ação.

A programação é feita através da interface de desenvolvimento MasterTool IEC XE (IDE). O MasterTool IEC XE possibilita o uso das seis linguagens no mesmo projeto, assim provendo as melhores características que cada linguagem pode oferecer ao usuário, resultando no desenvolvimento de aplicações mais eficientes, permitindo fácil documentação e manutenção futura.

Para mais informações sobre Programação, consultar Manual de Utilização MasterTool IEC XE - MU299609, Manual de Programação MasterTool IEC XE - MU399609 ou o padrão IEC 61131-3.

Organização e Acesso à Memória

A Série Nexto utiliza uma inovadora característica de organização e acesso à memória denominada big-endian, ou seja, o byte mais significativo é armazenado primeiramente e sempre será o de menor endereço (Ex.: %QB0 sempre será o mais significativo que o %QB1, como na Tabela 5-1, que utiliza a palavra UCPNEXTO, onde a letra U é o byte 0 e a letra O é o byte 7).

O acesso à memória deve ser realizado com cuidado pois, variáveis com maior número de bits (WORD, DWORD, LONG), utilizam como índice o byte mais significativo, ou seja, a %QD4 terá, como byte mais significativo, o byte %QB4. Não sendo necessário realizar cálculos para saber qual é a DWORD correspondente a determinados bytes. Organização little e big endian está na Tabela 5-1.

HSB <– Little-endian (Tradicional) –> LSB								
BYTE	**%QB7**	**%QB6**	**%QB5**	**%QB4**	**%QB3**	**%QB2**	**%QB1**	**%QB0**
	U	C	P	N	E	X	T	O
WORD	**%QW3**		**%QW2**		**%QW1**		**%QW0**	
	UC		PN		EX		TO	
DWORD	**%QD1**				**%QD0**			
	UCPN				EXTO			
LWORD	**%QL0**							
	UCPNEXTO							
HSB <– Big-endian (NEXTO) –> LSB								
BYTE	**%QB0**	**%QB1**	**%QB2**	**%QB3**	**%QB4**	**%QB5**	**%QB6**	**%QB7**
	U	C	P	N	E	X	T	O
WORD	**%QW0**		**%QW2**		**%QW4**		**%QW6**	
	UC		PN		EX		TO	
DWORD	**%QD0**				**%QD4**			
	UCPN				EXTO			
LWORD	**%QL0**							
	UCPNEXTO							

Tabela 5-1. Exemplo

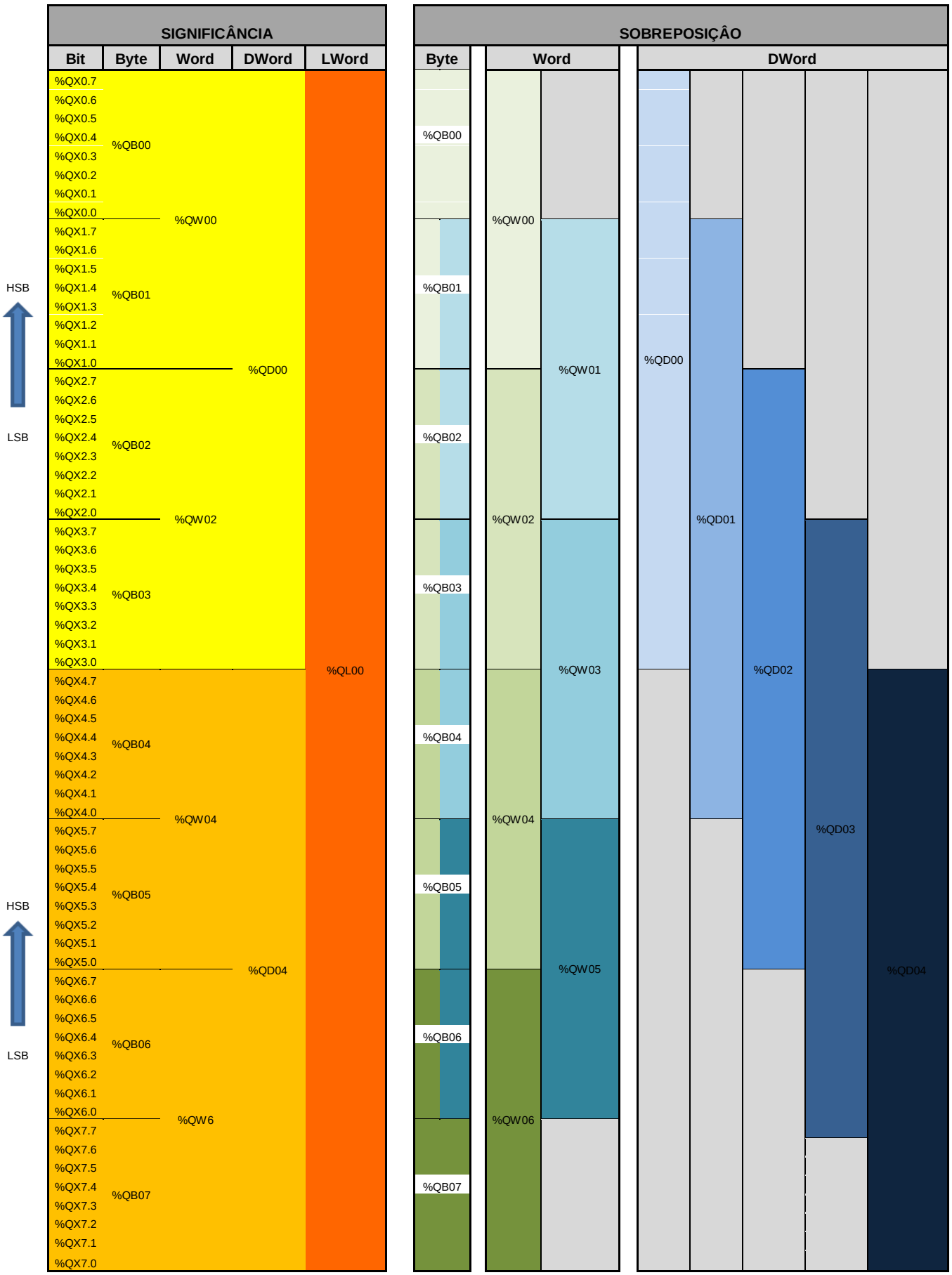

Tabela 5-2. Organização e Acesso à Memória

A Tabela 5-2 mostra a organização e acesso à memória, exemplificando a significância dos bytes (Significance) e a disposição dos demais tipos de variável, inclusive a sobreposição (Overlapping).

Perfis de Projeto

Um perfil de projeto no MasterTool IEC XE é um conjunto de regras, características comuns e padrões utilizados no desenvolvimento de uma solução de automação industrial, um perfil que influencia a forma de implementação da aplicação. Com a diversidade de tipos de aplicações suportadas pelo Runtime System (RTS) da Série Nexto, seguir um perfil é uma forma de reduzir a complexidade na programação. As aplicações podem ser criadas conforme um dos seguintes perfis:

- Simples
- Básico
- Normal
- Experiente
- Personalizado
- Perfil de Máquina

Para cada perfil definido para o RTS, o MasterTool IEC XE pode disponibilizar inúmeros templates compatíveis. Quando o usuário seleciona um template como modelo na criação de um projeto, a nova aplicação será desenvolvida conforme um determinado perfil, adotando as regras, características e padrões definidos pelo perfil associado ao template. Cada perfil de projeto define nomes padronizados para as tarefas e programas, os quais são pré-criados pelos templates de projeto. O desenvolvedor é obrigado a seguir rigorosamente a nomenclatura para tarefas, mas pode seguir ou não os nomes sugeridos para os respectivos programas.

Para garantir a compatibilidade de um projeto a um determinado perfil ao longo do desenvolvimento, são utilizadas duas abordagens:

- O MasterTool IEC XE somente permite a criação de projetos baseados em um template, selecionando ao mesmo tempo o perfil a ser utilizado.
- Na geração de código, o MasterTool IEC XE realiza a verificação de todas as regras definidas para o perfil válido para o projeto.

As próximas seções detalham as características ou padrões de cada perfil de projeto que seguem uma escala gradual de complexidade. Com base nestas definições, recomenda-se que o usuário sempre procure utilizar o perfil mais simples que atenda as necessidades da sua aplicação, migrando para outro mais sofisticado apenas quando as regras correspondentes estiverem sendo mais entraves ao desenvolvimento do que simplificações didáticas. Cabe ressaltar que a ferramenta de programação permite a alteração do perfil de um projeto existente (consulte a seção Atualização de projeto, no Manual de Utilização MasterTool IEC XE – MU299048), mas caberá ao desenvolvedor realizar qualquer ajuste necessário para que o projeto se torne compatível com as regras do novo perfil selecionado.

ATENÇÃO:
No decorrer dos perfis de projeto são nomeados alguns tipos de tarefas, as quais estão descritas na seção Configuração das Tarefas, no Manual de Utilização MasterTool IEC XE – MU299048.

ATENÇÃO:
Quando utilizado mais de uma tarefa, o acesso de E/S só deve ser executado no contexto da tarefa principal, MainTask. Caso não possa ser utilizada a opção *Habilita atualização de E/S por tarefa*, presente a partir da versão 2.01 do MasterTool IEC XE.

Simples

No perfil de projeto Simples, a aplicação possui apenas a tarefa de usuário MainTask. Esta tarefa é responsável pela execução de uma única unidade de programação do tipo Program denominada MainPrg. Este único programa pode chamar outras unidades de programação do tipo Programa, Função ou Bloco Funcional, mas todo código de usuário será executado exclusivamente pela tarefa MainTask.

Neste perfil, a tarefa MainTask será do tipo cíclica (Cyclic) com prioridade fixada como 13 (treze) e executa exclusivamente o programa MainPrg em um laço contínuo. A tarefa MainTask já está completamente definida e o desenvolvedor precisa criar o programa MainPrg optando por qualquer uma das linguagens da norma IEC 61131-3. Nem sempre é possível converter um programa para outra linguagem, mas sempre é possível criar um novo programa com o mesmo nome em substituição que seja construída em linguagem diversa. A opção padrão do MasterTool IEC XE é utilizar o MasterTool Standard Project associado ao perfil Single, o qual também inclui o programa MainPrg criado na linguagem escolhida na criação do projeto.

Uma aplicação deste tipo nunca precisa levar em consideração questões como consistência de dados, compartilhamento de recursos ou mecanismos de exclusão mútua.

Tarefa	POU	Prioridade	Tipo	Intervalo	Evento
MainTask	MainPrg	13	Cíclica	100 ms	-

Tabela 5-3. Tarefa no Perfil Simples

Básico

No perfil de projeto Básico, a aplicação possui uma tarefa de usuário do tipo Contínua denominada MainTask, que executa em um laço contínuo (sem definição de intervalo) com prioridade fixada como 13 (treze). Esta tarefa é responsável pela execução de uma única unidade de programação POU denominada MainPrg. É importante ressaltar que o intervalo pode variar em função da quantidade de tarefas de comunicação utilizadas, pois nesse modo, a tarefa principal é interrompida por tarefas de comunicação.

Este perfil também permite a inclusão de duas tarefas de evento com maior prioridade que podem interromper (preemptar) a MainTask a qualquer momento: a tarefa chamada ExternInterruptTask00 é uma tarefa de evento do tipo Extern com prioridade fixada em 02 (dois); a tarefa chamada TimeInterruptTask00 é uma tarefa de evento do tipo Cyclic com prioridade fixada como 01 (um).

O modelo de projeto Básico, inclui estas três tarefas já completamente definidas conforme apresentado na Tabela 5-4. O desenvolvedor precisa apenas criar os programas associados.

Tarefa	POU	Prioridade	Tipo	Intervalo	Evento
MainTask	MainPrg	13	Contínua	-	-
ExternInterruptTask00	ExternInterruptPrg00	02	Externa	-	IO_EVT_0
TimeInterruptTask00	TimeInterruptTask00	01	Cíclica	20 ms	-

Tabela 5-4. Tarefas no Perfil Básico

Normal

No perfil de projeto Normal, a aplicação possui uma tarefa de usuário do tipo Cyclic denominada MainTask. Essa tarefa é responsável pela execução de uma única unidade de programação do tipo Program denominada MainPrg. Este programa e esta tarefa são equivalentes a única tarefa e único programa do perfil Single, mas aqui a aplicação pode integrar tarefas adicionais de usuário. Estas outras tarefas são denominadas CyclicTask00 e CyclicTask01, cada qual responsável pela execução exclusiva do respectivo programa CyclicPrg<nn>. As tarefas CyclicTask<nn> são sempre do tipo cíclica (Cyclic) com prioridade fixada como 13 (treze), prioridade idêntica a tarefa MainTask. Estes

dois tipos formam um conjunto denominado de tarefas básicas, cujos programas associados podem chamar outras POUs do tipo Programa, Função ou Bloco Funcional.

Este perfil pode adicionalmente incluir tarefas de evento com maior prioridade que as tarefas básicas e que consequentemente podem interromper (preemptar) a execução daquelas a qualquer momento.

A tarefa chamada ExternInterruptTask00 é uma tarefa de evento do tipo Externa (Extern) cuja execução é disparada por algum evento externo, tais como variação de um sinal de controle em uma porta serial ou variação de uma entrada digital no barramento do NEXTO. A prioridade desta tarefa é fixada em 02 (dois), sendo responsável pela execução exclusiva do programa ExternInterruptPrg00. A tarefa chamada TimeInterruptTask00 é uma tarefa de evento do tipo Cíclica (Cyclic) com prioridade fixada como 01 (um), sendo responsável pela execução exclusiva do programa TimeInterruptPrg00.

No modelo de projeto Normal, existem cinco tarefas, e suas POUS, já completamente definidas conforme apresentado na Tabela 5-5. O desenvolvedor precisa apenas implementar os conteúdos dos programas optando, no wizard, por qualquer uma das linguagens da norma IEC 61131-3. Os intervalos e eventos de disparo das tarefas podem ser configurados pelo desenvolvedor e as tarefas desnecessárias deverão ser eliminadas.

Tarefa	POU	Prioridade	Tipo	Intervalo	Evento
MainTask	MainPrg	13	Cíclica	100 ms	-
CyclicTask00	CyclicPrg00	13	Cíclica	200 ms	-
CyclicTask01	CyclicPrg01	13	Cíclica	500 ms	-
ExternInterruptTask00	ExternInterruptPrg00	02	Externa	-	IO_EVT_0
TimeInterruptTask00	TimeInterruptTask00	01	Cíclica	20 ms	-

Tabela 5-5. Tarefas no Perfil Normal

Experiente

O perfil de projeto Experiente inclui as mesmas tarefas básicas, MainTask, CyclicTask<nn>, ExternInterruptTask00 e TimeInterruptTask00, com as mesmas prioridades (13; 02 e 01 respectivamente), mas é uma expansão dos anteriores, pois admite múltiplas tarefas de evento. Ou seja, a aplicação pode incluir várias tarefas ExternInterruptTask<nn> ou TimeInterruptTask<nn> executando os programas ExternInterruptPrg<nn> e TimeInterruptPrg<nn>. As prioridades das tarefas de evento adicionais podem ser livremente selecionadas na faixa de 08 a 12. Neste perfil, além dos programas padrões, cada tarefa pode executar programas adicionais.

Neste perfil de projeto, a aplicação ainda pode incluir a tarefa de usuário FreeTask do tipo Freewheeling com prioridade 31, responsável pela execução do programa FreePrg. Como esta tarefa é de baixa prioridade pode ser interrompida por todas as demais logo a mesma é capaz de executar códigos que podem ficar bloqueados.

Existem oito tarefas já completamente definidas conforme apresentado na Tabela 5-6, bem como os respectivos programas associados criados na linguagem que o usuário selecionar. Os intervalos e eventos de disparo de qualquer tarefa e as prioridades das tarefas de evento podem ser configurados também pelo usuário.

Ao desenvolver a aplicação usando o perfil de projeto Experiente, é necessário um cuidado especial com o escalonamento das tarefas de evento. Caso exista compartilhamento de informações e recursos entre estas tarefas ou entre estas e as tarefas básicas é fortemente recomendável adotar estratégias para garantir a consistência de dados.

Tarefa	POU	Prioridade	Tipo	Intervalo	Evento
MainTask	MainPrg	13	Cíclica	100 ms	-
CyclicTask00	CyclicPrg00	13	Cíclica	200 ms	-
CyclicTask01	CyclicPrg01	13	Cíclica	500 ms	-
ExternInterruptTask00	ExternInterruptPrg00	02	Externa		IO_EVT_0

TimeInterruptTask00	TimeInterruptTask00	01	Cíclica	20 ms	-
ExternInterruptTask01	ExternInterruptPrg01	11	Externa	-	IO_EVT_1
TimeInterruptTask01	TimeInterruptPrg01	09	Cíclica	30 ms	-
FreeTask	FreePrg	31	Contínua	-	-

Tabela 5-6. Tarefas no Perfil Experiente

Personalizado

O perfil de projeto Personalizado permite ao desenvolvedor explorar todas as potencialidades do Runtime System implantado nas unidades centrais de processamento da Série Nexto. Nenhuma das funcionalidades é desabilitada, nenhuma prioridade, associação entre tarefas e programas ou nomenclatura é imposta. A única exceção se faz para a MainTask que deve sempre existir com este nome neste Perfil.

Além das tarefas em tempo real com prioridades 00 a 15, as quais são escalonadas por prioridade, neste perfil também é possível definir tarefas com prioridades menores na faixa 16 a 31. Nesta faixa, é usado o Completely Fair Scheduler (time sharing), o que é necessário para execução de códigos que podem ficar bloqueados (por exemplo, uso de sockets).

O desenvolvedor tem a liberdade para seguir parcialmente ou não a organização definida nos demais perfis de projeto, conforme as particularidades de sua aplicação. Por outro lado, o modelo personalizado, associado a este perfil não necessita elementos pré-definindo como tarefa, programa ou parâmetro, cabendo ao desenvolvedor a criação de todos os elementos que compõe a aplicação. Entretanto o usuário pode gerar os mesmos elementos disponíveis para o perfil Experiente.

Tarefa	POU	Prioridade	Tipo	Intervalo	Evento
MainTask	MainPrg	13	Cíclica	100 ms	-
CyclicTask00	CyclicPrg00	13	Cíclica	200 ms	-
CyclicTask01	CyclicPrg01	13	Cíclica	500 ms	-
ExternInterruptTask00	ExternInterruptPrg00	02	Externa	-	IO_EVT_0
TimeInterruptTask00	TimeInterruptTask00	01	Cíclica	20 ms	-
ExternInterruptTask01	ExternInterruptPrg01	11	Externa	-	IO_EVT_1
TimeInterruptTask01	TimeInterruptPrg01	09	Cíclica	30 ms	-
FreeTask	FreePrg	31	Contínua	-	-

Tabela 5-7. Tarefas no Perfil Personalizado

Perfil de Máquina

No perfil de Máquina, por padrão, a aplicação possui uma tarefa de usuário do tipo Cíclica denominada MainTask. Esta tarefa é responsável pela execução de uma única unidade de programação do tipo Program denominada MainPrg. Este programa pode chamar outras unidades de programação do tipo Program, Function ou Function Block, mas todo o código do usuário será executado exclusivamente pela tarefa MainTask.

Este perfil se caracteriza por permitir intervalos menores na tarefa MainTask, permitindo a execução mais rápida do código do usuário. Esta otimização é possível pois a MainTask executa também o processamento do barramento. Desta forma diferente de outros perfis, o perfil de máquina não necessita de chaveamento de contexto para o tratamento do barramento o que reduz o tempo de processamento geral.

Este perfil pode adicionalmente incluir uma tarefa de interrupção de tempo, denominada TimeInterruptTask00, com maior prioridade que a tarefa MainTask e que consequentemente pode interromper a execução dela a qualquer momento.

Tarefa	POU	Prioridade	Tipo	Intervalo	Evento
MainTask	MainPrg	13	Cíclica	20 ms	-
TimeInterruptTask00	TimeInterruptTask00	01	Cíclica	4 ms	-

Tabela 5-8. Tarefas no Perfil de Máquina

Tabela Geral

		Perfis de Projeto					
Verificações		Simples	Máquina	Básico	Normal	Experiente	Personalizado
Total de tarefas	**Quantidade**	01	02	[01..03]	[01..32]	[01..32]	[01..32]
Programas por tarefa	**Quantidade**	01		01	01	<n>	<n>
Main Task	**Nome**						
	Tipo	Cíclica	Cíclica	Contínua	Cíclica	Cíclica	<n>
	Prioridade	13	13	13	13	13	<n>
	Quantidade	01	01	01	01	01	01
Time Interrupt Task	**Nome**						<n>
	Tipo		Cíclica	Cíclica	Cíclica	Cíclica	Cíclica
	Prioridade		01	01	01	01 ou [08..12]	<n>
	Quantidade		[00..01]	[00..01]	[00..01]	[00..31]	[00..31]
Extern Interrupt Task	**Nome**						<n>
	Tipo			Externa	Externa	Externa	Externa
	Prioridade			02	02	02 ou [08..12]	<n>
	Quantidade			[00..01]	[00..01]	[00..31]	[00..31]
Ciclic Task	**Nome**				CyclicTask<nn>	CyclicTask<nn>	<n>
	Tipo				Cíclica	Cíclica	Cíclica
	Prioridade				13	13	<n>
	Quantidade				[00..31]	[00..31]	[00..31]
Free Task	**Nome**					FreeTask	<n>
	Tipo					Roda livre	Roda livre
	Quantidade					31	<n>
	Quantidade					[00..01]	[00..01]
Event Task	**Nome**						<n>
	Tipo						Evento
	Prioridade						<n>
	Quantidade						[00..31]

Tabela 5-9. Tabela Geral de Perfis x Tarefas

ATENÇÃO:
Os nomes sugeridos para as POUs associadas às tarefas não são consistidos. Os mesmos podem ser substituídos desde que sejam substituídos também nas configurações das tarefas.

Número Máximo de Tarefas

O número máximo de tarefas que o usuário poderá criar somente está definido para o perfil Personalizado, ou seja, o único que tem essa permissão. Os demais já têm as suas tarefas criadas e configuradas. No entanto, as tarefas que forem criadas precisam utilizar os prefixos, de acordo com o tipo de cada uma das tarefas, a seguir: CyclicTaskxx, TimeInterruptTaskxx, ExternInterruptTaskxx, onde xx representa o número da tarefa que está sendo criada.

A Tabela 5-10 descreve a quantidade máxima de tarefas IEC por UCP e perfil de projeto, sendo que as instâncias de protocolo também são consideradas tarefas de comunicação pela UCP.

	Tipo de Tarefa	NX3004						NX3010						NX3020						NX3030					
		S	B	N	E	P	M	S	B	N	E	P	M	S	B	N	E	P	M	S	B	N	E	P	M
Configuration Task (Tarefa WHSB)	**Cíclica**	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0
Tarefas de Usuário	**Cíclica**	1	1	15	15	15	2	1	1	15	15	15	2	1	1	23	23	23	2	1	1	31	31	31	2
	Disparada por Evento	0	0	0	0	15	0	0	0	0	0	15	0	0	0	0	0	23	0	0	0	0	0	31	0
	Disp. Evento Externo	0	1	1	14	15	0	0	1	1	14	15	0	0	1	1	22	23	0	0	1	0	30	31	0
	Contínua	0	1	0	1	15	0	0	1	0	1	15	0	0	1	0	1	23	0	0	1	0	1	31	0
	Disparada por Estado	0	0	0	0	15	0	0	0	0	0	15	0	0	0	0	0	23	0	0	0	0	0	31	0
NETs – Instâncias Cliente ou Servidor	**Cíclica**	4						4						8						16					
COM (n) – Instâncias Mestre ou Escravo	**Cíclica**	1						1						1						1					
TOTAL		16						16						24						32					

Tabela 5-10. Número Máximo de Tarefas IEC

Notas:

Legenda dos Perfis: As letras S, B, N, E, P e M corresponde respectivamente aos Perfis Simples, Básico, Normal, Experiente, Personalizado, Máquina.

Valores: Os números definidos para cada tipo de tarefa representam os valores máximos permitidos.

Tarefa WHSB: A tarefa WHSB que é uma tarefa de sistema deve ser considerada para que não seja ultrapassado o valor total.

NETs: Instâncias Cliente ou Servidor: O valor máximo definido considera todas as interfaces Ethernet do sistema, ou seja, inclui os módulos de expansão, quando estes são aplicados. Exemplos para esse tipo de tarefa são as instâncias do Protocolo MODBUS.

COM (n): Instâncias Mestre ou Escravo: O "n" representa o número da interface serial, ou seja, mesmo com módulos de expansão, o valor da tabela será o máximo por interface. Exemplos para esse tipo de tarefa são as instâncias do Protocolo MODBUS.

Total: O valor total não representa a soma de todas as tarefas por perfil, mas o valor máximo permitido por UCP. Então, o usuário poderá criar vários tipos de tarefas, desde que o número estabelecido para cada uma e o valor total, não sejam ultrapassados.

Configurando a UCP

A configuração da UCP Nexto baseia-se em estruturar a área de diagnósticos, a área de memória retentiva e persistente, modo de troca a quente, entre outros parâmetros.

O usuário deverá dar dois cliques na UCP Nexto, localizada na árvore de dispositivos, conforme mostra a Figura 5-1, e configurar os campos conforme descrito no capítulo Configuração da UCP.

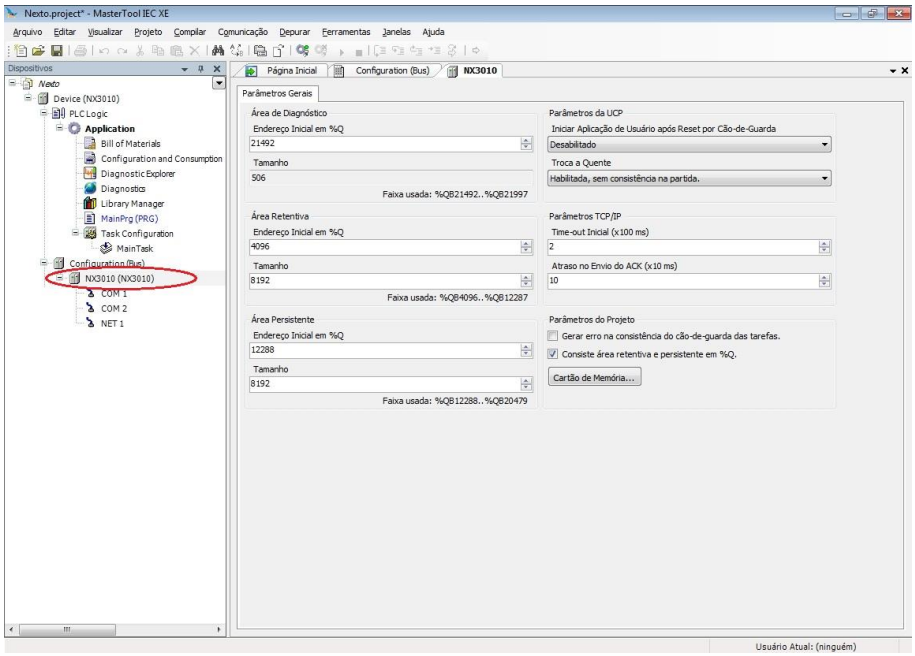

Figura 5-1. Configuração da UCP

Além disso, para que seja possível a comunicação entre a UCP e o MasterTool IEC XE, a interface Ethernet NET 1 deverá ser configurada, conforme o capítulo Configuração das Interfaces Ethernet – NET 1.Clicando duas vezes sobre o ícone NET 1 da UCP, na árvore de dispositivos, surgirá uma nova aba para configuração da rede de comunicação em que o módulo está conectado.

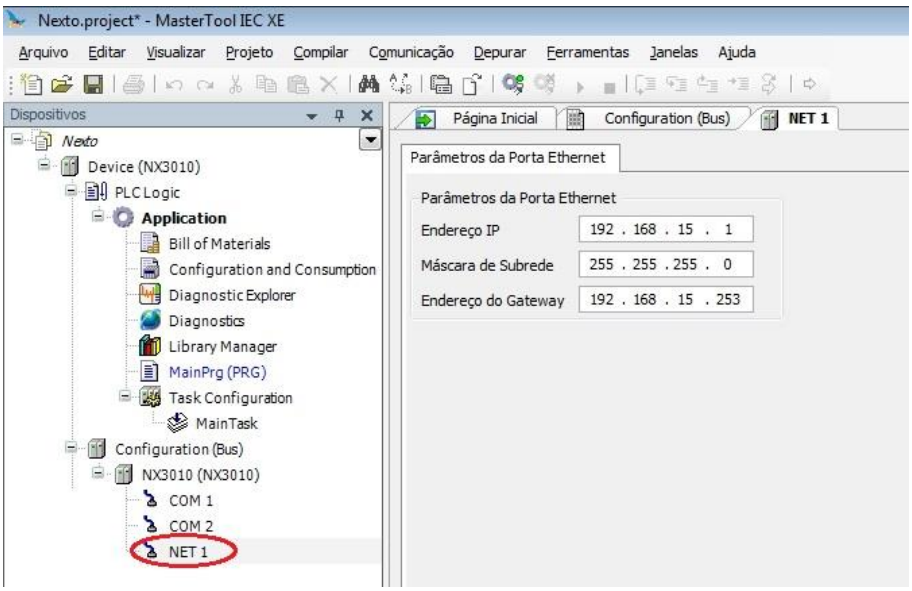

Figura 5-2. Configurando a Porta de Comunicação da UCP

Caso a UCP com o IP configurado não seja encontrada na rede ou a UCP que está atualmente ativa tenha um IP diferente da ativa configurada no projeto, uma mensagem surgirá na tela durante o

Login, solicitando ao usuário a possibilidade de substituir o IP antigo pelo configurado (opção *Sim*) ou escolher a opção *Não* para não enviar o projeto.

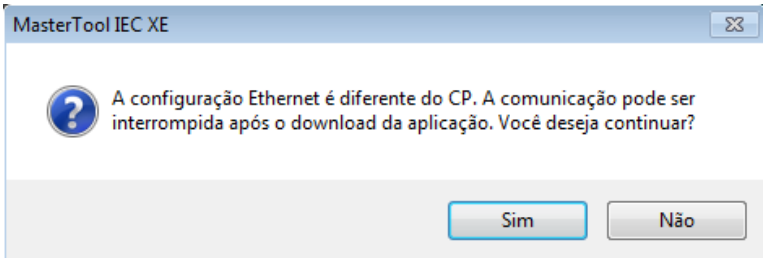
MasterTool IEC XE
A configuração Ethernet é diferente do CP. A comunicação pode ser interrompida após o download da aplicação. Você deseja continuar?
Sim
Não

Figura 5-3. Aviso de configuração IP

Bibliotecas

Existem diversos recursos da ferramenta de programação que estão disponíveis através de bibliotecas. Sendo assim, os mesmos devem ser inseridos no projeto para que a sua utilização seja possível. O procedimento de inserção, bem como mais informações sobre as bibliotecas disponíveis podem ser encontrados no Manual de Programação IEC 61131 – MP399048, capítulo Bibliotecas.

Inserindo uma Instância de Protocolo

As próprias UCPs da Série Nexto, conforme o capítulo Características Gerais, disponibilizam protocolos, como o MODBUS. Basta adicionar e configurar (conforme o capítulo Configuração de Protocolos) a instância do protocolo desejado na interface de comunicação.

Abaixo, são descritos dois casos de inserção do protocolo MODBUS, sendo um na interface serial e outro na interface Ethernet.

MODBUS RTU

O primeiro passo para configurar o MODBUS RTU, em modo escravo, é incluir a instância na COM desejada (nesse caso, COM 1). Clicar com o botão direito sobre a COM e selecionar “Acrescentar Dispositivo...”, conforme mostra a Figura 5-4:

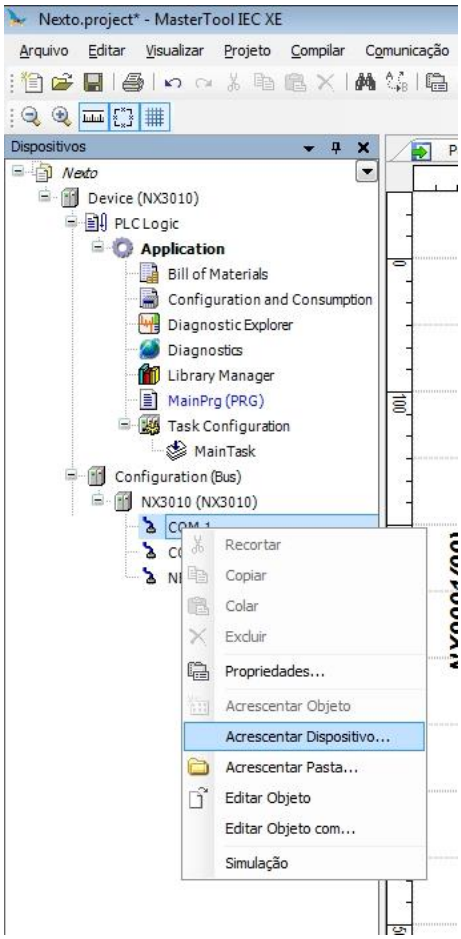

Figura 5-4. Adicionando a Instância

Após, surgirão na tela os protocolos disponíveis ao usuário. Definir o modo de configuração do protocolo, selecionar a opção “MODBUS Symbol RTU Slave”, para configuração por Mapeamento Simbólico ou “MODBUS RTU Slave”, para configuração por Representação Direta (%Q). A seguir, clicar em Acrescentar, conforme mostra a Figura 5-5.

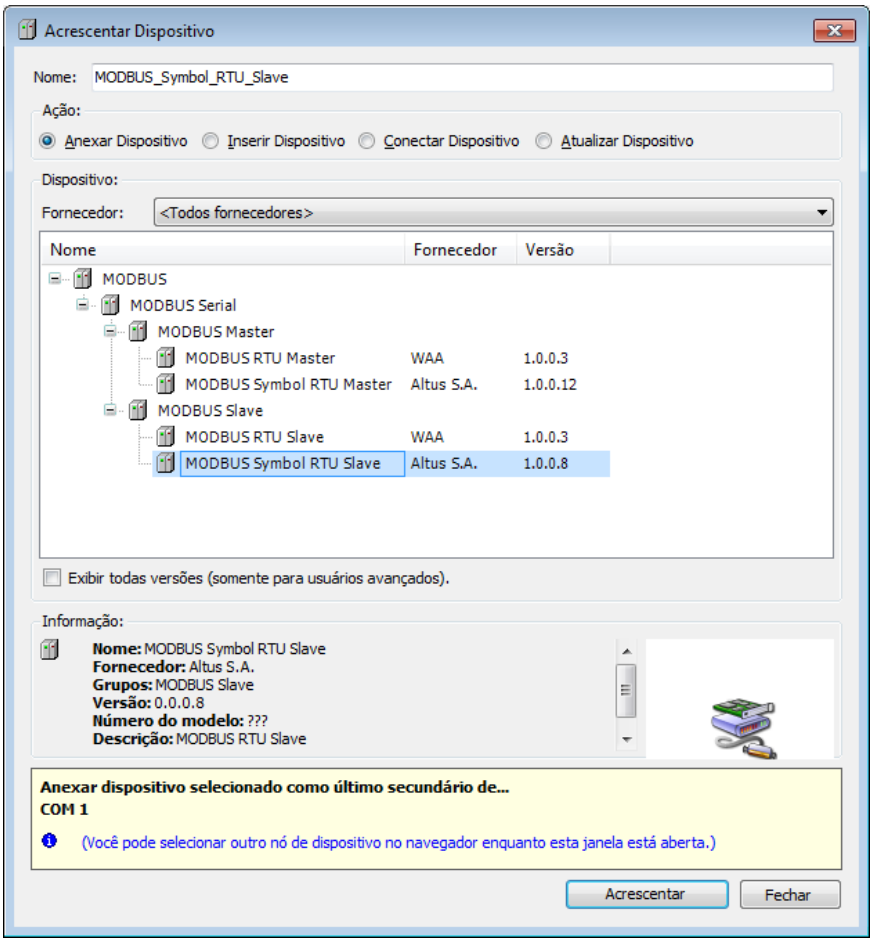

Figura 5-5. Selecionando o Protocolo

MODBUS Ethernet

O primeiro passo para configurar o MODBUS Ethernet, em modo cliente, é incluir a instância na NET desejada (nesse caso NET 1, pois a UCP NX3010 possui uma interface Ethernet). Clicar com o botão direito sobre a NET e selecionar “Acrescentar Dispositivo...”, conforme mostra a Figura 5-6.

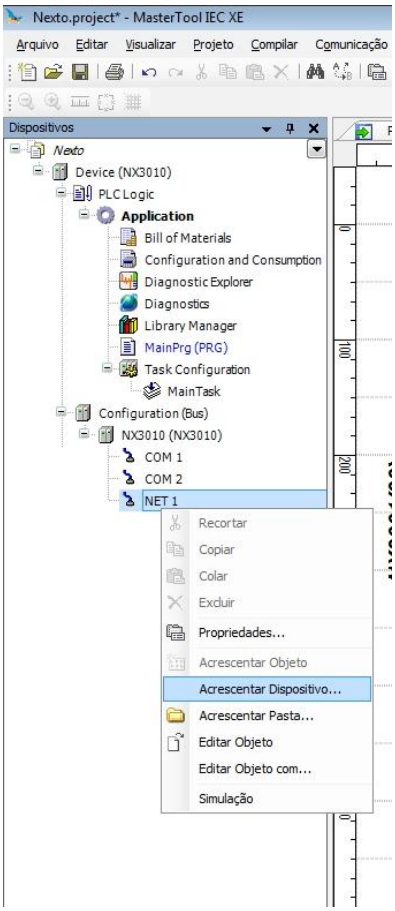

Figura 5-6. Adicionando a Instância

Após irá surgir na tela os protocolos disponíveis ao usuário. Definir o modo de configuração do protocolo, selecionar a opção “MODBUS Symbol Client”, para configuração por Mapeamento Simbólico, ou “MODBUS Client”, para configuração por Representação Direta (%Q). A seguir, clicar em Acrescentar, conforme mostra a Figura 5-7.

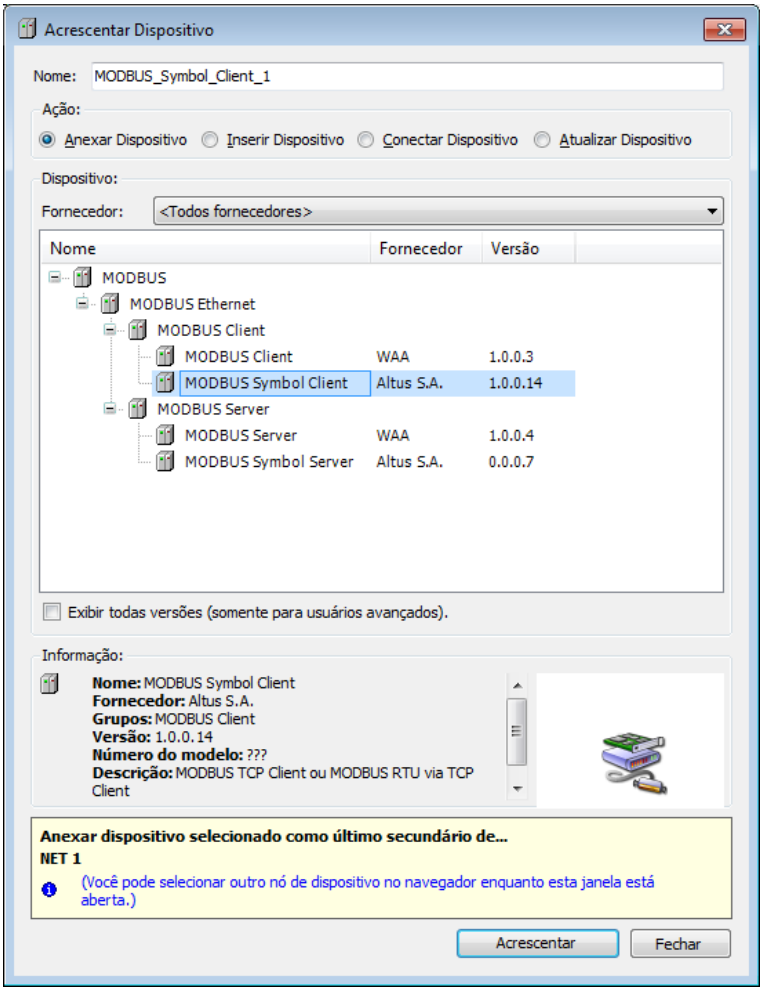

Figura 5-7. Selecionando o Protocolo

Localizando a Rede

Como existe a possibilidade de mais UCPs estarem conectadas à rede, o usuário deve localizar todas as unidades de comunicação e selecionar a desejada.

Inicialmente, será necessário acessar a opção Device, na árvore de dispositivos, clicando duas vezes. Na aba “Configurações de Comunicação”, selecionar o Gateway, ou, caso nenhum Gateway exista ou o usuário queira adicionar um novo Gateway, clicar no botão “Acrescentar Gateway”, configurando seu IP na janela que será aberta. Para mapear os dispositivos presentes na rede, deve-se então clicar no botão “Mapear Rede".

Então, deve-se aguardar enquanto o software MasterTool IEC XE procura e apresenta as UCPs disponíveis na rede.

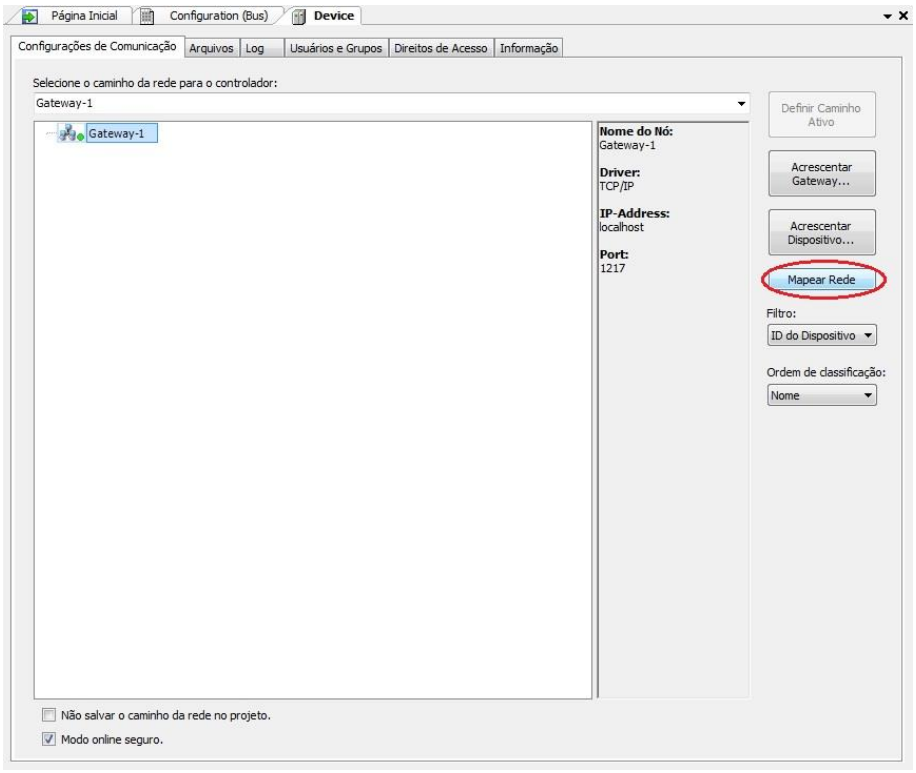

Figura 5-8. Localizando a UCP

Após, basta selecionar a UCP desejada e clicar na opção “Definir Caminho Ativo”, fazendo com que ela fique ativa e o configurador saiba com quem deve comunicar e enviar o projeto.

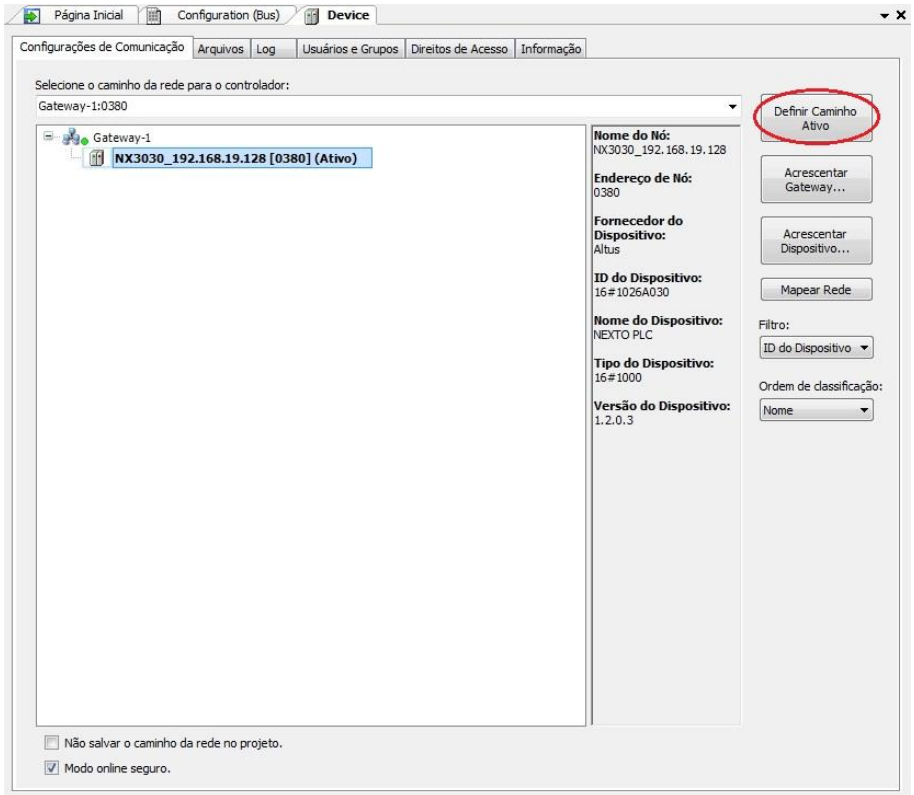

Figura 5-9. Ativando a UCP

Caso seja necessário, o usuário pode alterar o nome padrão do dispositivo que é exibido. Para isso, deve clicar com o botão direito do mouse sobre o dispositivo desejado e selecionar a opção “Alterar Nome do Dispositivo”. Após uma mudança de nome, o dispositivo não voltará ao nome padrão em nenhuma circunstância.

Login

Após compilar a aplicação e corrigir todos os erros encontrados, é o momento de enviar o projeto para a UCP. Para que isso seja possível, basta efetuar a operação de Login no software MasterTool IEC XE. Essa operação pode levar alguns segundos, dependendo do tamanho do arquivo gerado. Para executar o Login, basta acessar o menu Comunicação e clicar na opção “Login”, conforme mostra o exemplo da Figura 5-10.

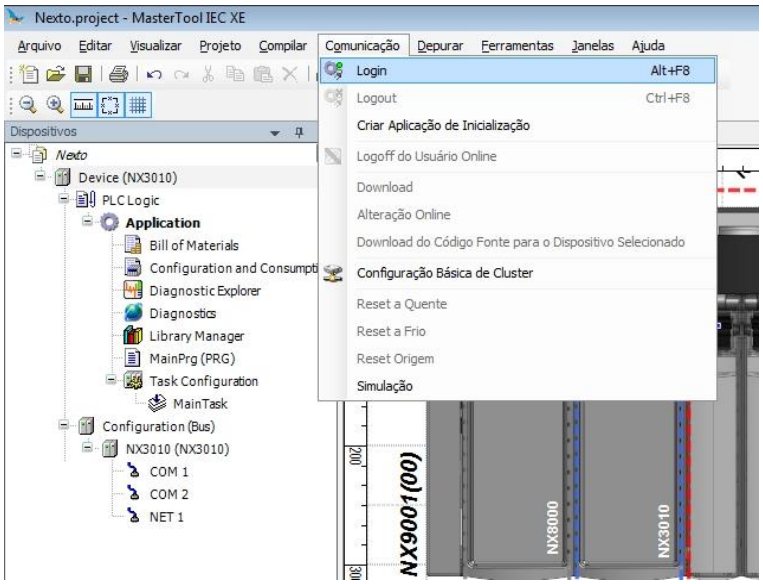

Figura 5-10. Enviando o Projeto para a UCP

Após a execução do comando, poderão surgir algumas mensagens de interface com o usuário, as quais são apresentadas devido a diferença entre um projeto antigo e o que está sendo enviado ou, simplesmente, alteração no valor de alguma variável.

A Figura 5-11 mostra a mensagem que o MasterTool IEC XE irá apresentar em caso do novo projeto, que está sendo enviado seja diferente de um projeto já existente dentro da UCP. As opções disponíveis são as seguintes:

- Login com alteração online: executar o login e enviar o novo projeto sem que a aplicação atual da UCP seja parada (ver item Modo Run), atualizando as alterações quando um novo ciclo é executado.
- Login com Download: executar o login e enviar o novo projeto com a UCP parada (ver item Modo Stop). Quando a aplicação for iniciada, as atualizações já terão sido realizadas.
- Login sem alteração: executa o login sem enviar o novo projeto.

ATENÇÃO: Antes da versão 2.01 do MasterTool IEC XE, quando o Login com alteração online era executado, a aplicação não ficava salva na memória de programa. Era necessário executar o comando “Criar Aplicação de Inicialização” no menu Comunicação, sem efetuar o logout, para que a aplicação fosse gravada na memória de programa. A partir da versão 2.01 esta operação passou a ser executada de forma automática sem a necessidade de executar o comando.

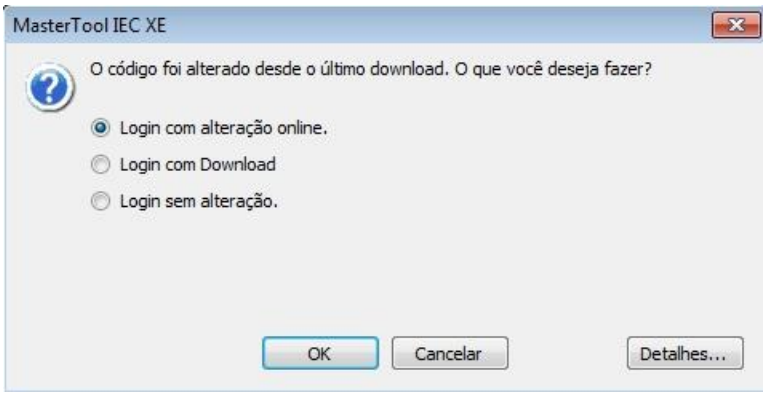

Figura 5-11. Atualização do Projeto na UCP

ATENÇÃO Nas alterações online não é permitido associar mapeamentos de variáveis simbólicas de uma lista de variáveis globais (GVL) e utilizar essas variáveis em outra lista de variáveis globais (GVL).

A Figura 5-12 mostra a mensagem que o MasterTool IEC XE exibe quando somente foram realizadas alterações em variáveis da aplicação, fazendo com que não seja possível enviar o novo projeto e o mesmo ser atualizado em um novo ciclo da UCP, a qual está em execução (ver item Modo Run). Dessa forma, o MasterTool IEC XE solicita ao usuário se o login deve ser executado como download e a UCP parada (ver item Modo Stop) ou a operação deve ser cancelada.

Obs.: O botão "Detalhes..." apresenta as modificações realizadas na aplicação.

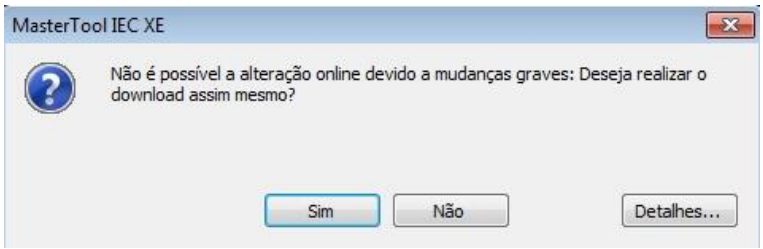
MasterTool IEC XE
Não é possível a alteração online devido a mudanças graves: Deseja realizar o download assim mesmo?
Sim
Não
Detalhes...

Figura 5-12. Alteração de Variáveis

Caso seja o primeiro envio de aplicação para a UCP, a mensagem da Figura 5-13 irá surgir na tela do MasterTool IEC XE.

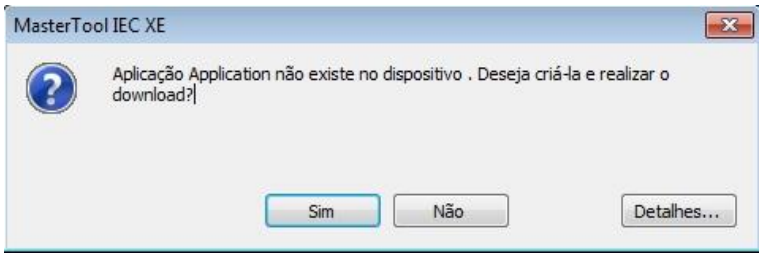

Figura 5-13. Primeiro Envio de Aplicação

Modo Run

Logo após enviar o projeto para a UCP, a aplicação não será executada imediatamente (somente se for realizada uma alteração online), antes que seja selecionado um novo comando "Iniciar". Essa função possibilita ao usuário controlar a execução da aplicação enviada para a UCP. Além disso, permite que valores iniciais sejam pré-configurados, para que, no primeiro ciclo já sejam atualizados na UCP.

Para selecionar tal funcionalidade, basta acessar o menu Depurar e clicar em “Iniciar”, conforme mostra a Figura 5-14.

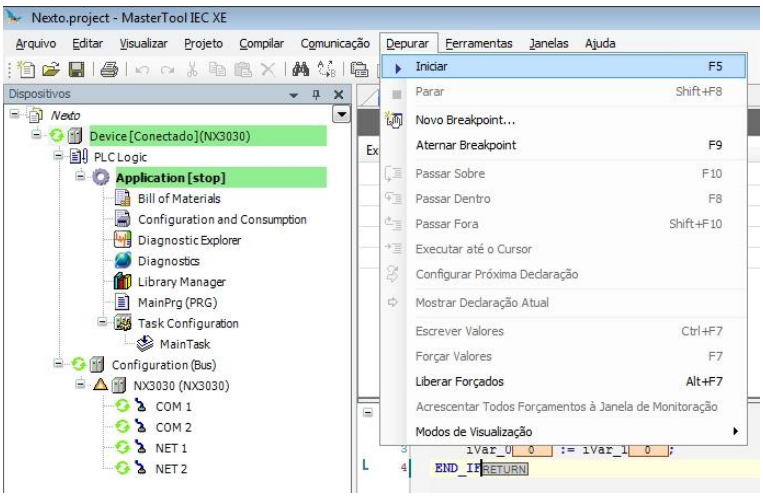

Figura 5-14. Iniciando a Aplicação

Abaixo, a Figura 5-15 mostra a aplicação em execução. Caso seja selecionada a aba de uma POU, as variáveis criadas serão listadas em uma janela de monitoração, na qual valores podem ser forçados e visualizados pelo usuário. Caso as variáveis sejam forçadas através do comando F7 do teclado, a UCP indicará essa condição no visor gráfico. Maiores detalhes consultar capítulo Visor Gráfico.

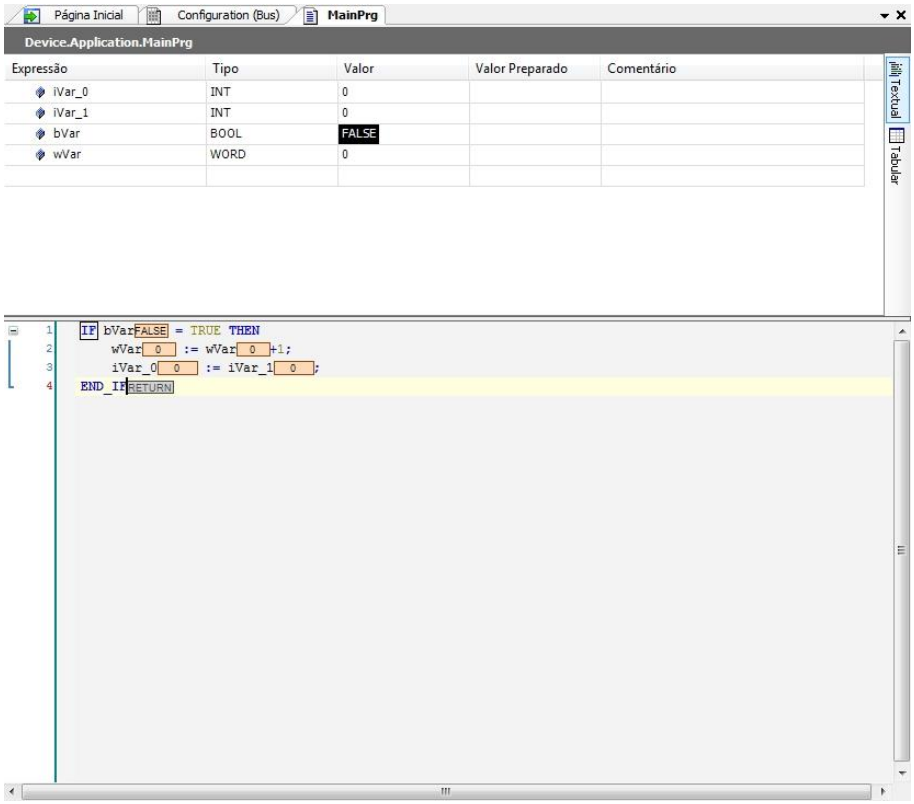

Figura 5-15. Programa em Execução

Caso a UCP é inicializada como uma aplicação já gravada internamente, ela automaticamente entra em Modo Run, sem a necessidade de realizar o comando via MasterTool IEC XE.

Modo Stop

Para interromper a execução da UCP, sem que o software MasterTool IEC XE perca a conexão com a mesma, o usuário deve selecionar a opção Parar, disponível no menu Depurar, “Parar”, conforme mostra a Figura 5-16.

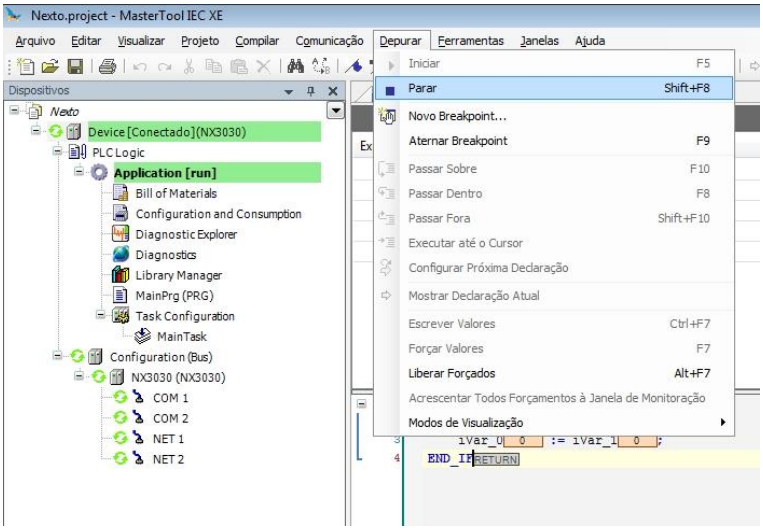

Figura 5-16. Parando a Aplicação

Caso a UCP seja inicializada sem aplicação gravada, ela automaticamente entra em Modo Stop, assim como quando ocorre uma exceção de software.

Escrita e Forçamento de Variáveis

Após efetuar um Login em um CP, o usuário poderá escrever ou forçar valores em variáveis do projeto.

O comando de escrita (Ctrl + F7) escreve um valor em uma variável, podendo este valor ser sobrescrito pelas instruções executadas na aplicação.

Já um comando de escrita forçada (F7) irá escrever um valor na variável, sem permitir que este valor seja alterado até que sejam liberadas as variáveis forçadas.

É importante ressaltar que, quando utilizados os protocolos MODBUS RTU Escravo e MODBUS Ethernet Servidor e a opção “Somente Leitura” das relações configuradas não estiver selecionada, o comando de escrita forçada (F7) deve ser realizado sobre as variáveis disponíveis na janela de monitoração, pois o comando de escrita (Ctrl + F7) deixa as variáveis serem sobrescritas quando realizadas novas leituras.

ATENÇÃO: O forçamento de variáveis pode ser realizado somente em modo Online na UCP. Variáveis de diagnóstico não podem ser forçadas, apenas escritas, pois diagnósticos são providos pela UCP e devem poder ser sobrescritos pela mesma.

Quando for efetuada uma escrita forçada em uma variável redundante do CP Ativo, a MainTask da aplicação sofrerá um impacto em seu tempo de execução, tanto no CP Ativo, quando no CP Reserva. Isto porque os dois Half-Clusters irão trocar a cada ciclo informações sobre as variáveis forçadas. Portanto, quando for forçar variáveis em um sistema redundante, deve-se considerar o acréscimo que pode ter a execução da tarefa. A Tabela 5-11 exemplifica em quanto será acrescida, em média, a execução da MainTask quando isto ocorrer:

	CP Ativo			CP Reserva		
Tempo de Execução	**50 ms**	**100 ms**	**200 ms**	**50 ms**	**100 ms**	**200 ms**
Acréscimo com 10 forçamentos	2,4 %	2,2 %	1,7 %	4,0 %	3,4 %	2,0 %
Acréscimo com 50 forçamentos	12,0 %	9,2 %	6,0 %	18,0 %	12,0 %	8,0 %
Acréscimo com 128 forçamentos	26,0 %	21,0 %	16,0 %	56,0 %	34,0 %	22,5 %

Tabela 5-11. Influência do Forçamento de Variáveis em um CP Redundante

ATENÇÃO:
Quando uma UCP está com variáveis forçadas e é desenergizada, na próxima inicialização as variáveis perderão o forçamento.
O limite de forçamentos para as UCPs Nexto é de 128 variáveis, independente do modelo de UCP ou configuração utilizada.

Logout

Caso a opção do usuário seja finalizar a comunicação com a UCP, deve ser utilizado comando "Logout", localizado no menu Comunicação, "Logout", conforme mostra a Figura 5-17.

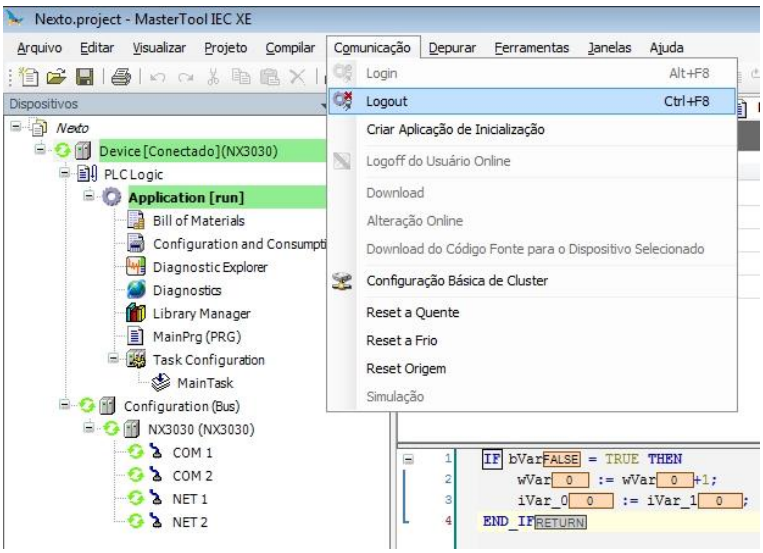

Figura 5-17. Interrompendo a Comunicação com a UCP

Upload do Projeto

As UCPs da Série Nexto possibilitam a gravação de um projeto na memória do produto, o qual pode ser recuperado e reutilizado através do software MasterTool IEC XE.

Para armazenar um projeto na memória do produto, a UCP deve estar conectada (Login) e a opção de envio de projeto, juntamente com o aplicativo, deve estar selecionada.

Para recuperar o projeto previamente armazenado, deve ser selecionado as opções, conforme mostra a Figura 5-18.

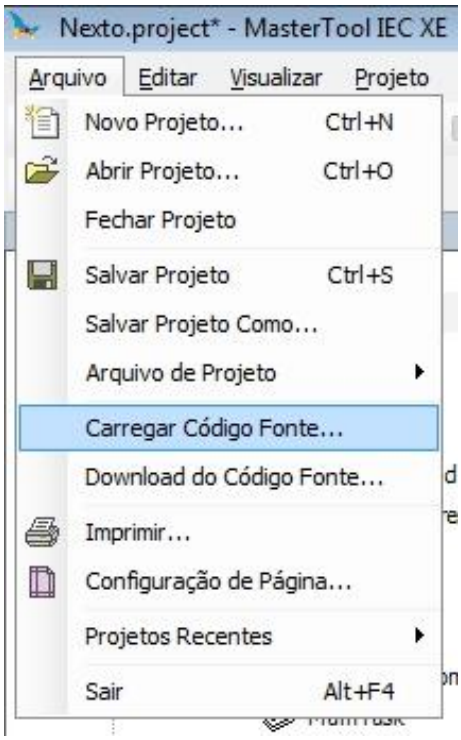

Figura 5-18. Opção de Upload de Projeto

Após, basta selecionar a UCP desejada e clicar em OK, conforme a Figura 5-19.

Para garantir que o projeto carregado da UCP seja completamente igual e possa ser carregado sem problemas a partir de outras estações, consulte o capítulo Método de Envio/Login de Projetos Sem Diferença de Projetos do Manual de Utilização MasterTool IEC XE MT8500 – MU299048.

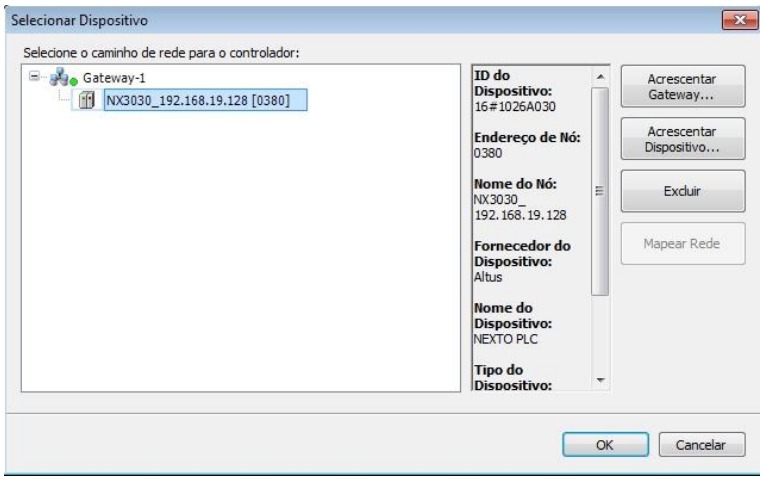

Figura 5-19. Seleciona a UCP

ATENÇÃO: O tamanho da área de memória para armazenar um projeto nas UCPs Nexto está definido na Tabela 2-5.

ATENÇÃO:
O Upload recupera o último projeto armazenado no controlador conforme descrito nos parágrafos anteriores. Caso ocorra apenas o carregamento para execução de um determinado aplicativo, o mesmo não poderá ser recuperado pelo procedimento de Upload.

Estados de Operação da UCP

Run

Quando uma UCP está em modo Run indica que todas as tarefas da aplicação estão em execução.

Stop

Quando uma UCP está em modo Stop, indica que as tarefas da aplicação estão paradas. O valor das variáveis nas tarefas são mantidas com o valor atual e variáveis de saída assumem valores definidos pelo usuário.

Quando uma UCP passa para modo Stop a partir do envio de uma aplicação, as variáveis nas tarefas da aplicação serão perdidas exceto as variáveis do tipo persistente. As variáveis de saída assumirão o valor definido pelo usuário e em seguida o valor das saídas passará para o estado seguro. Assim que a nova aplicação for carregada as variáveis de saída assumirão novamente o valor definido pelo usuário.

Breakpoint

Quando uma marca de depuração é atingida em uma tarefa, essa tarefa é interrompida. Todas as demais tarefas ativas na aplicação não irão ser interrompidas, elas continuarão a sua execução. Neste modo é possível percorrer um programa no modo Online. Um passo a passo pode ser executado e as posições das interrupções de depuração dependem do editor.

Para maiores informações sobre a utilização de marcas de depuração, favor consultar o Manual de Utilização do MasterTool IEC XE – MU299048.

Exception

Quando uma UCP está em Exception indica que alguma operação indevida ocorreu em uma das tarefas ativas da aplicação. A tarefa que causou o Exception será suspensa e as demais tarefas irão para o modo Stop. Somente é possível tirar as tarefas desse estado e colocá-las em execução novamente após uma nova condição de partida da UCP. Portanto somente com um Reset a Quente, Reset a Frio, Reset Origem ou uma reinicialização da UCP coloca novamente a aplicação em modo Run.

Reset a Quente

Este comando coloca a UCP em modo Stop e inicializa todas as variáveis das tarefas da aplicação, exceto as variáveis dos tipos retentiva e persistente. As variáveis inicializadas com um valor específico assumirão exatamente este valor, as demais variáveis assumirão o valor padrão de inicialização (zero).

Reset a Frio

Este comando coloca a UCP em modo Stop e inicializa todas as variáveis das tarefas da aplicação, exceto as variáveis do tipo persistente. As variáveis inicializadas com um valor específico assumirão exatamente este valor, as demais variáveis assumirão o valor padrão de inicialização (zero).

Reset Origem

Este comando remove todas as variáveis das tarefas da aplicação, inclusive as variáveis do tipo persistente e apaga a aplicação da UCP.

Notas:

Reset: Quando um Reset é executado os breakpoints definidos na aplicação são desabilitados.

Comando: Para executar um comando de qualquer tipo de Reset é necessário estar em modo Online na UCP.

6. Redundância com UCP NX3030

Introdução

Este capítulo aborda a redundância de CPs da Série Nexto, que pode ser utilizada somente com a UCP NX3030.

Trata-se de uma redundância do tipo hot-standby, onde os controladores são duplicados. Um dos controladores normalmente está em estado Ativo e controlando o processo, enquanto o outro controlador normalmente está em estado Reserva, mantendo-se sincronizado com o controlador Ativo. Caso ocorra uma falha no controlador Ativo, que o impeça de continuar controlando o processo, o controlador Reserva chaveia automaticamente para Ativo, em um tempo suficientemente baixo para não perturbar o processo, sem causar descontinuidades nas saídas que controlam o processo.

A redundância hot-standby é um método utilizado para aumentar a tolerância a falhas e, consequentemente, aumentar a disponibilidade do sistema de automação. A ideia básica é que nenhuma falha simples em componentes duplicados cause a interrupção do controle do processo.

A redundância hot-standby é muito aplicada em:

- plataformas de exploração de petróleo
- sistemas de geração e distribuição de energia
- intertravamentos de segurança (Sistemas Instrumentados de Segurança)
- processos contínuos, tais como plantas químicas, refinarias de petróleo, produção de celulose, etc.

Além dos controladores, podem ser também duplicadas, opcionalmente, as redes de campo (PROFIBUS DP e Ethernet), as redes de supervisão Ethernet, e as redes de controle Ethernet HSDN (High Speed Deterministic Network). Optando-se pela duplicação destas redes, obtém-se uma disponibilidade ainda maior.

A redundância hot-standby de CPs da Série Nexto não prevê duplicação de módulos de E/S. Caso a redundância de módulos de E/S seja desejável, ela pode ser tratada em nível de aplicação, pelo usuário final. Por exemplo, o usuário pode duplicar ou até mesmo triplicar um módulo de entradas analógicas, e criar um algoritmo de votação para determinar qual das entradas irá ser considerada em determinado momento em sua aplicação.

A Figura 6-1 mostra um exemplo típico de arquitetura redundante com a UCP NX3030.

A parte central de um CP redundante é formada por dois bastidores idênticos, denominados CPA e CPB, além de um painel de controle de redundância (PX2612), quando esta opção for configurada. No contexto da redundância, cada bastidor (CPA e CPB) é denominado half-cluster, enquanto o conjunto formado por estes dois bastidores é denominado cluster.

Neste exemplo, observam-se também duplicações em uma rede de campo PROFIBUS, na rede Ethernet de supervisão, e na rede de controle Ethernet HSDN.

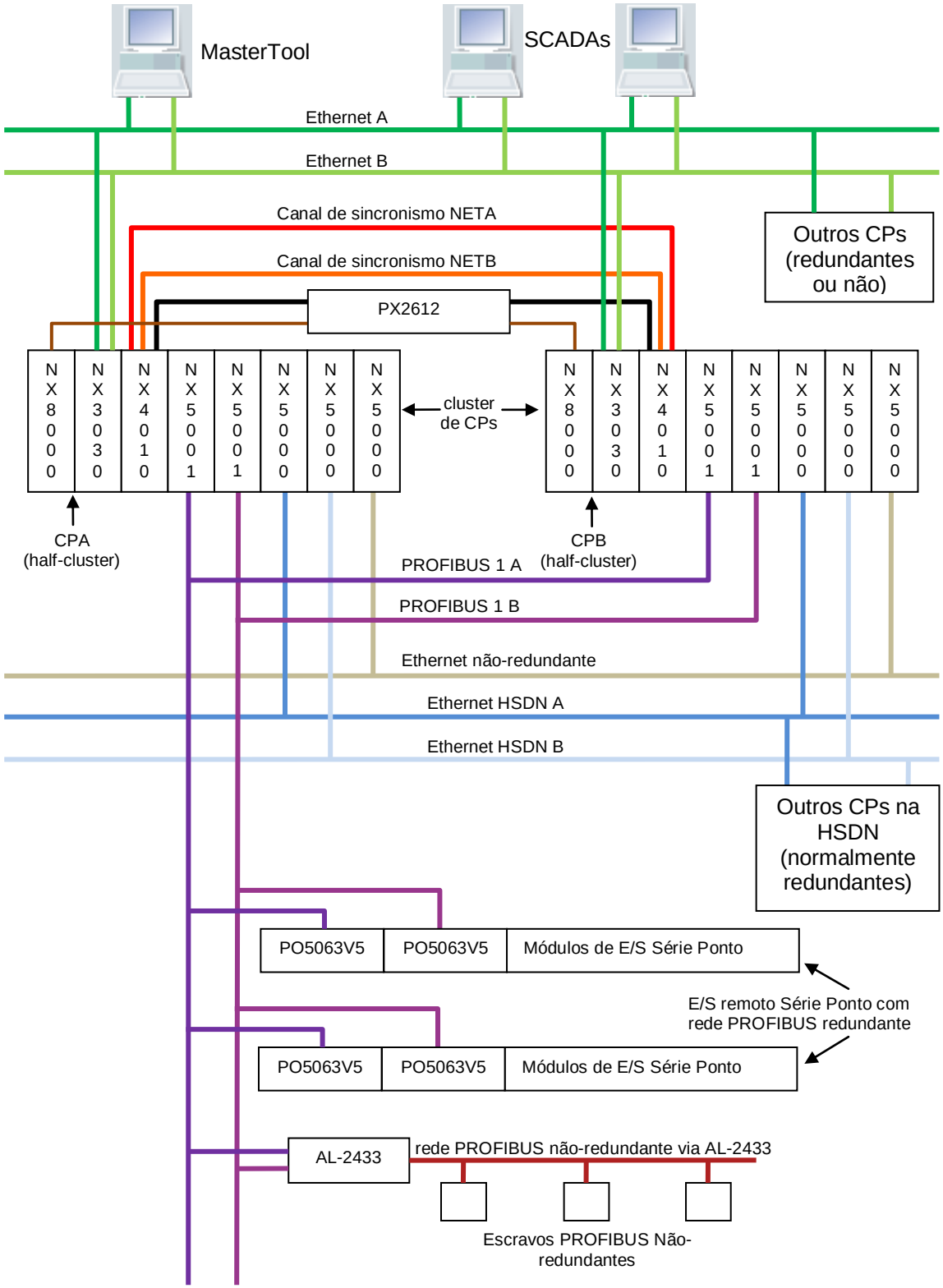

Figura 6-1. Exemplo de Arquitetura Redundante com UCP NX3030

Descrição Técnica e Configuração

Configuração Mínima de um CP Redundante (Sem utilização do Painel PX2612)

Um CP redundante é composto, no mínimo, de:

- dois half-clusters idênticos

Cada half-cluster é constituído, no mínimo, dos seguintes módulos:

- o próprio bastidor onde os módulos são inseridos, que pode ser um dos seguintes:
  - NX9000, com 8 posições
  - NX9001, com 12 posições
  - NX9002, com 16 posições
  - NX9003, com 24 posições
- a fonte de alimentação NX8000, nas posições 0 e 1 do bastidor
- a UCP NX3030, nas posições 2 e 3 do bastidor
- o módulo NX4010, nas posições 4 e 5 do bastidor

A Figura 6-2 mostra um exemplo de configuração mínima de um CP redundante, utilizando o menor bastidor (NX9001, com 12 posições). Neste caso, observa-se que os 3 módulos inseridos no bastidor têm largura dupla (ocupam duas posições do bastidor).

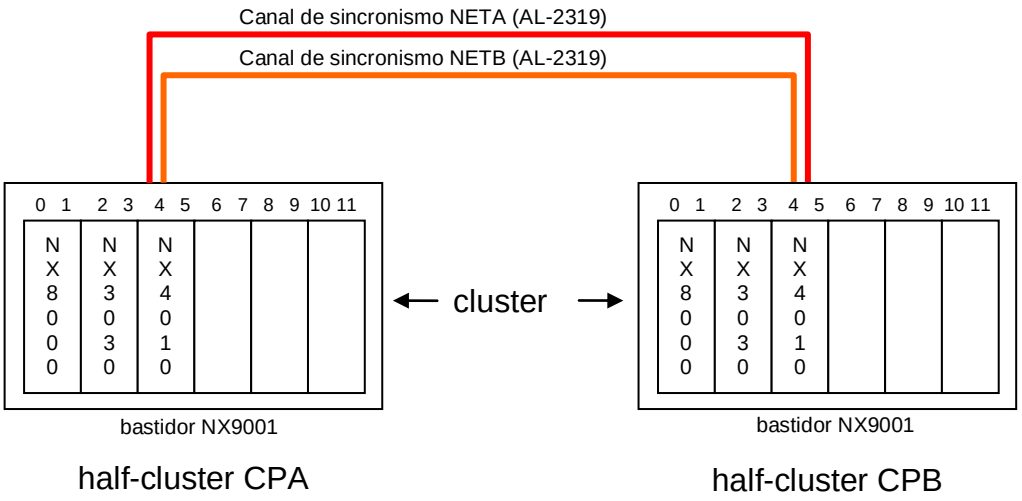

Figura 6-2. Configuração Mínima de um CP Redundante em Bastidor NX9001

Configurações Típicas de um CP Redundante

Uma configuração mínima, como esta mostrada na Figura 6-2,já pode ser útil como uma “unidade de processamento redundante”. Seria possível utilizar os canais de comunicação seriais e Ethernet da UCP NX3030, por exemplo, para comunicação MODBUS TCP com um sistema SCADA, e comunicação MODBUS RTU e/ou MODBUS TCP com dispositivos de campo inteligentes ou sistemas de E/S remoto MODBUS.

Em configurações mais típicas, no entanto, são inseridos módulos adicionais nos half-clusters CPA e CPB, por exemplo, para prover um sistema de E/S remoto PROFIBUS e canais Ethernet adicionais.

Entre os módulos adicionais que, opcionalmente, podem ser inseridos nos half-clusters, pode-se citar:

- Mestres PROFIBUS NX5001
- Interfaces Ethernet NX5000

Caso seja necessário, pode-se utilizar bastidores maiores, como NX9002 (16 posições) e NX9003 (24 posições). Deve-se observar que todos os módulos citados até o momento neste capítulo têm largura dupla (ocupam duas posições).

Adicionalmente, também pode ser utilizado o painel PX2612, que permite executar algumas transições na máquina de estados redundante que, caso contrário, não seriam possíveis, além do desligamento automático de half-clusters em algumas situações de falha.

Adição de Módulos NX5001 para Redes PROFIBUS

É possível inserir até quatro módulos NX5001 para a utilização de redes PROFIBUS em um CP redundante. Cada uma destas redes pode ser simples ou redundante. Caso a rede PROFIBUS "n" (sendo "n" um número de 1 a 4) seja redundante, as duas redes que a compõem são nomeadas PROFIBUS "n" A e PROFIBUS "n" B. Caso a rede PROFIBUS "n" seja simples, a rede que a compõe é nomeada PROFIBUS "n" A.

Para criar uma rede PROFIBUS redundante, é necessário inserir dois módulos NX5001 em cada half-cluster. Para criar uma rede PROFIBUS simples, basta inserir um módulo NX5001 em cada half-cluster. Sendo assim, é possível configurar até quatro redes simples, ou duas redes redundantes, ou uma redundante e duas simples. Nos demais casos, menos do que quatro módulos NX5001 serão necessários em cada half-cluster. Mais informações sobre redes PROFIBUS são fornecidas na seção Princípios de Funcionamento - Configurações de Redes PROFIBUS.

No exemplo da Figura 6-1, existe apenas uma rede PROFIBUS (PROFIBUS 1), e a mesma é redundante (PROFIBUS 1 A e PROFIBUS 1 B). Neste exemplo, portanto, foram inseridos dois módulos NX5001 em cada half-cluster.

Adição de Módulos NX5000 para Redes Ethernet

É possível inserir até seis módulos NX5000 em cada half-cluster, provendo seis canais Ethernet adicionais, além dos dois canais Ethernet já existentes na UCP NX3030.

Os canais Ethernet podem ser usados de forma individual, ou organizados em pares NIC Teaming. Pares NIC Teaming são utilizados para prover canais Ethernet redundantes e serão descritos com maiores detalhes na seção Princípios de Funcionamento - Redes Ethernet Redundantes com NIC Teaming.

Uma aplicação típica para o módulo NX5000 pode ser construir uma HSDN (High Speed Deterministic Network), para comunicação entre diversos CPs redundantes. Através desta rede, diversos CPs redundantes podem trocar mensagens em uma rede totalmente segregada, para garantir determinismo e uma comunicação rápida. Além disso, configurando esta rede como redundante com pares NIC Teaming, pode-se obter excelente disponibilidade. Para construir uma HSDN redundante, deve-se inserir dois módulos NX5000 em cada half-cluster. A Figura 6-1 mostra um exemplo de HSDN redundante, utilizando dois módulos NX5000 em cada half-cluster.

Aplicações nas quais os módulos de entradas e saídas estejam conectados a redes Ethernet podem necessitar interfaces extras de módulos NX5000 para se conectar a estas redes. Nestes casos, a rede que conecta os módulos de entradas e saídas pode ser uma rede simples ou redundante. Além disso, as interfaces podem ser configuradas com a opção de gerar falha vital. Neste caso, uma falha na rede provocará um switch-over.

A Figura 6-1 também mostra um exemplo com um módulo NX5000 usado de forma isolada (sem redundância NIC Teaming), inserido à direita dos demais módulos em cada bastidor.

Módulo NX4010

O módulo NX4010, como mostra a Figura 6-3, foi concebido para prover a interconexão entre os dois half-clusters CPA e CPB, e também para conectar estes half-clusters ao painel de controle de redundância PX2612. Maiores detalhes sobre as conexões destes módulos serão fornecidos na seção Interconexões entre Half-Clusters e Painel de Controle de Redundância PX2612.

Figura 6-3. Módulo NX4010

Características NX4010

Suas principais características são:

- Sincronismo de dados e aplicação entre dois half-clusters
- Interface de comunicação redundante entre dois half-clusters
- Switchover automático (troca do half-cluster ativo) em caso de time-out de comunicação entre NX4010 e a sua respectiva UCP
- Possibilidade de desligar o outro half-cluster
- One Touch Diag
- Electronic Tag on Display
- Visor e LEDs para indicação de diagnóstico

Demais características (gerais, elétricas, mecânicas e de ambiente) estão presentes nas Características Técnicas do Módulo de Redundância NX4010 – CT114900.

Painel de Controle de Redundância PX2612

O painel de controle PX2612 é um item opcional em um sistema redundante. Ele deve ser utilizado quando a opção de redundância com painel for selecionada na criação de projeto usando o wizard. A Figura 6-4 mostra uma fotografia do painel de controle de redundância PX2612, enquanto a Figura 6-5 mostra com mais detalhes seu painel frontal.

Através do conector DB9 denominado CONTROL PLC A, é feita a conexão ao conector CONTROL do NX4010 do PLCA, utilizando o cabo AL-2317/A.

Através do conector DB9 denominado CONTROL PLC B, é feita a conexão ao conector CONTROL do NX4010 do PLCB, utilizando o cabo AL-2317/B.

Além disso, existe um conector com cinco bornes machos:

- GND: borne para conexão de aterramento.

- RL A: 2 bornes que correspondem aos terminais de um relé NA (normalmente aberto), que pode ser comandado pelo CPB para desligar o CPA. Este relé deve ser fechado pelo CPB para desligar o CPA.
- RL B: 2 bornes que correspondem aos terminais de um relé NA (normalmente aberto), que pode ser comandado pelo CPA para desligar o CPB. Este relé deve ser fechado pelo CPA para desligar o CPB.

Um CP (CPA ou CPB) poderá desligar o outro CP (CPB ou CPA) em algumas situações excepcionais, utilizando, então, os relés NA nos conectores RL A e RL B. Tais situações são descritas na seção Transições entre Estados de Redundância.

O PX2612 também dispõe de seis botões para comandos de redundância, e seis LEDs utilizados para indicações de estado de redundância. Cada CP lê três destes botões e controla três destes LEDs.

Maiores informações sobre as funções destes botões e LEDs serão fornecidas na seção Funções do Painel de Comando de Redundância PX2612.

Figura 6-4. Painel de Controle de Redundância PX2612

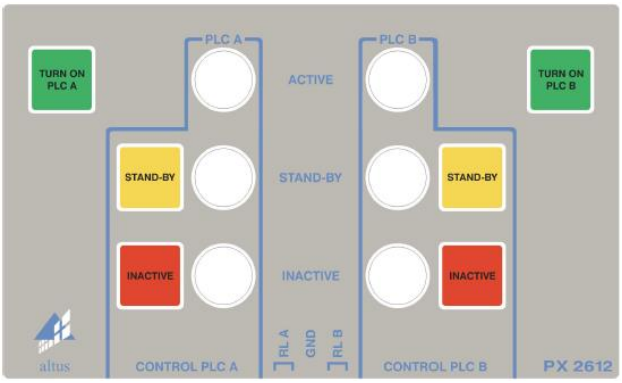

Figura 6-5. Frontal do Painel de Controle de Redundância PX2612

Características PX2612

O painel de controle de redundância PX2612 possui as seguintes características:

- CONTROL PLC A: conexão ao módulo NX4010 do CPA
- CONTROL PLC B: conexão ao módulo NX4010 do CPB
- RL A: terminais de um relé NA utilizado para desligar o CPA
- RL B: terminais de um relé NA utilizado para desligar o CPB
- GND: aterramento

Demais características (gerais, elétricas, mecânicas e de ambiente) estão presentes nas Características Técnicas do Painel de Controle de Redundância PX2612 – CT112500.

Interconexões entre Half-Clusters e Painel de Controle de Redundância PX2612

A Figura 6-6 mostra como devem ser feitas as conexões entre CPA, CPB e PX2612, inclusive para possibilitar que um CP desligue o outro, o que é necessário em situações excepcionais.

Dois cabos AL-2319 devem ser utilizados para os canais de sincronismo de redundância NETA e NETB. Um destes cabos interconecta os conectores NET 1 dos NX4010 de cada CP (canal NETA). O outro cabo interconecta os conectores NET 2 dos NX4010 de cada CP (canal NETB).

Um cabo AL-2317/A interconecta o conector CONTROL do NX4010 do CPA ao conector CONTROL PLC A do PX2612.

Um cabo AL-2317/B interconecta o conector CONTROL do NX4010 do CPB ao conector CONTROL PLC B do PX2612.

Além disso, é necessário fazer um circuito especial de alimentação, para que um CP seja capaz de desligar o outro CP em casos extremos.

Para maior confiabilidade, deve-se utilizar duas fontes separadas de 24V, uma para energizar o CPA, outra para energizar o CPB.

Observa-se que é necessário utilizar dois relés externos do tipo normalmente fechado (NF), com capacidade de corrente suficiente para energizar o NX8000. Estes relés devem ser dimensionados para uma corrente nominal da ordem de 2 A. Diodos em paralelo com as bobinas dos relés NF devem ser utilizados para proteger os contatos dos relés NA do PX2612.

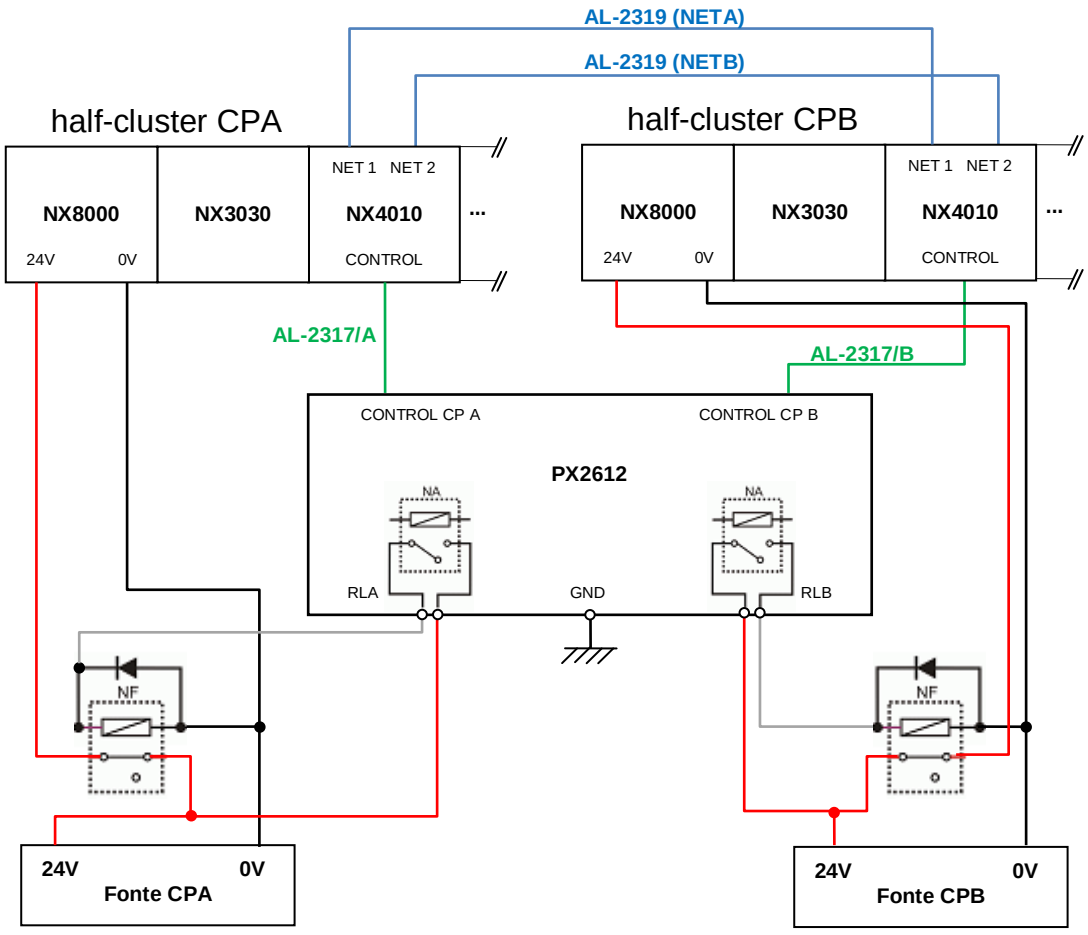

Figura 6-6. Interconexões entre CPA, CPB e PX2612

Características Gerais

	Características Gerais de um CP redundante
UCPs permitidas	NX3030
Tipo de redundância	Hot-standby
Tolerância a falhas	Tolera, no mínimo, falha simples em equipamentos duplicados nos half-clusters. Em casos específicos, pode tolerar múltiplas falhas.
5 estados de redundância de um half-cluster	- Não-Configurado: estado inicial, também considerado quando CP está desligado ou não está executando a tarefa principal (MainTask). - Inicializando: estado temporário assumido depois de Não-Configurado, onde alguns testes determinarão o próximo estado (Inativo, Ativo, Reserva – ou de volta para Não-Configurado). - Inativo: estado atingido depois de determinados tipos de falhas ou para manutenção programada. - Ativo: controlando o processo do usuário. - Reserva: apto para chavear para Ativo e controlar o processo do usuário, caso haja uma demanda (ex: falha do CP Ativo).
Principais falhas que causam switchover entre CP Ativo e CP Reserva. **CP Reserva chaveia para Ativo e o Ativo pode ir para Inativo ou Não-Configurado.**	- Falta de alimentação. - Fonte. - UCP (parada de execução da tarefa principal - MainTask). - NX4010. - Falha de ambos os canais de sincronismo (NETA e NETB) sem que a causa seja interna ao CP Reserva. Neste caso, o CP Reserva, além de assumir como Ativo, desliga o outro CP. - Falha de ambos os canais de sincronismo (NETA e NETB) sendo a causa interna ao CP Ativo. - Falha em alguma rede PROFIBUS configurada como vital. - Falha em alguma rede Ethernet configurada como vital.
Comandos que causam switchover entre CP Ativo e CP Reserva	- Comandos via painel de controle de redundância (PX2612). - Comandos recebidos do MasterTool ou de um sistema SCADA, através deste CP (local) ou do outro CP (remoto). - Comandos gerados pela aplicação do usuário, por exemplo, em função de outros diagnósticos (ex: falha de comunicação Ethernet), através deste CP (local) ou do outro CP (remoto).
Principais falhas que impedem um CP de ficar no estado Reserva, ou assumir o estado Reserva. **Tais falhas fazem com que este CP assuma os estados Não-Configurado ou Inativo.**	- Falta de alimentação. - Fonte. - UCP (parada de execução da tarefa principal - MainTask). - NX4010. - Falha de ambos os canais de sincronismo (NETA e NETB) onde a causa é interna ao CP Reserva. - Falha no serviço de sincronismo de dados redundantes. - Falha no serviço de sincronismo da lista de forçamentos redundantes. - Falha total em alguma rede PROFIBUS configurada como vital. - Falha total em alguma rede Ethernet configurada como vital. - Projeto diferente daquele do CP Ativo, com sincronização automática de projetos habilitada. - Versão de firmware incompatível com CP Ativo. - Falha de Interface Ethernet configurada como falha vital.
Comandos que tiram o CP do estado Reserva	- Comandos via painel de controle de redundância (PX2612). - Comandos recebidos do MasterTool ou de um sistema SCADA, através deste CP (local) ou do outro CP (remoto). - Comandos gerados pela aplicação do usuário, por exemplo, em função de outros diagnósticos (ex: falha de comunicação Ethernet), através deste CP (local) ou do outro CP (remoto).
Tempo de switchover	- Tipicamente de um a três ciclos da MainTask, dependendo do estímulo para troca de estado (comando ou falha). - Em caso de falha na rede PROFIBUS simples, dois ciclos da MainTask + 500 ms.
Switchover sem descontinuidades (bump-less)	- Um switchover não provoca descontinuidades nas saídas do controlador, nem em variáveis internas.
Overhead da redundância (consumo de UCP a cada ciclo	- Valor máximo calculado automaticamente pelo MasterTool e informado ao usuário, considerando uma lista de forçamentos

da MainTask adicionado pela redundância)	redundantes vazia. - Valor médio típico de 60 ms para 224 Kbytes de dados redundantes, em um sistema com uma rede PROFIBUS redundante e duas redes Ethernet HSDN redundantes
Visor da UCP	Entre outros diagnósticos, mostra o estado da redundância (Ativo, Reserva, Inativo, Não-Configurado, inicializando). Para a identificação do CP, é necessário verificar no menu Redundancia, onde há a identificação do CPA ou CPB.
Painel de controle de redundância PX2612	- Este item é opcional. - Através de botões, permite comandos de switchover ou transições entre estados de redundância para manutenção. - LEDs sinalizam o estado da redundância em cada half-cluster. - Relé NA permite que um half-cluster desligue o outro em casos extremos. Botão permite o religamento do outro half-cluster. - Recursos embutidos para teste dos botões, LEDs e relés. - Um projeto criado com painel não pode ser convertido em um projeto sem painel, e vice-versa.
Diagnósticos da redundância	- Indicam falhas tanto no CPA como no CPB, independente dos seus estados (Ativo ou Não-Ativo). - Evitam “falhas ocultas”. - Permitem manutenção rápida, o que é essencial para obter alta disponibilidade.
Comandos da redundância	- Permitem executar as mesmas ações do painel de controle PX2612, além de outros comandos (ex: comandar switchovers). - Podem ser executados no CP local, ou repassados para o outro CP (remoto) via canais de sincronismo NETA / NETB. - Podem ser recebidos a partir do MasterTool ou de um sistema SCADA. - Podem ser executados a partir da aplicação do usuário.
Eventos da redundância	Registram modificações nos diagnósticos e comandos de redundância, com estampa de tempo, permitindo assim uma investigação das causas de um switchover.
SNTP (Simple Network Time Protocol)	Permite que os eventos tenham estampa de tempo precisa e ajustada à hora mundial. Também sincroniza o relógio de tempo real do CP para outras aplicações.
Sincronização de comandos e diagnósticos	A cada ciclo da MainTask, CPA e CPB trocam diagnósticos e comandos através dos canais de sincronismo NETA ou NETB. Desta maneira, um CP conhece os diagnósticos e comandos do outro.
Sincronização de dados redundantes	A cada ciclo da MainTask, o CP Ativo copia dados redundantes para o CP Não-Ativo através dos canais de sincronismo NETA e NETB. Dados não-redundantes não são sincronizados.
Sincronização da lista de forçamentos redundantes	A cada ciclo da MainTask, o CP Ativo copia a lista de forçamentos redundantes para o CP Não-Ativo através dos canais de sincronismo NETA e NETB. Esta lista inclui apenas variáveis redundantes forçadas. Desta maneira, CPA e CPB poderão ter diferentes conjuntos de dados não-redundantes forçados, pois estes forçamentos não são sincronizados.
Projeto único para CPA e CPB	Existe um projeto único comum para o CPA e CPB, gerado pelo MasterTool. O projeto é composto do projeto aplicativo (código executável) e do project archive (código fonte).
Identificação de um CP	Através do MasterTool, identifica-se uma UCP NX3030 como CPA, CPB ou CP não-redundante. Esta identificação não faz parte do projeto aplicativo gerado pelo MasterTool, embora seja escrita em um CP usando o MasterTool. A identificação de cada CP viabiliza a característica de ter um projeto único para CPA e CPB.
Sincronização automática do projeto	Se o projeto do CP Ativo se tornar diferente do projeto do CP Não-Ativo, o mesmo é copiado do CP Ativo para o CP Não-Ativo. Esta sincronização pode durar diversos ciclos da MainTask. Deve-se lembrar que o projeto é composto do projeto aplicativo (código executável) e do código fonte (project archive), sendo ambos sincronizados. Esta sincronização pode ser desabilitada em circunstâncias especiais, para viabilizar modificações de projeto que só poderiam ser carregadas off-line em CPs não-redundantes.
Expansão a quente de módulos e remotas PROFIBUS	Existem modificações de projeto que não podem ser feitas à quente (on-line) em um CP não-redundante, tais como a inclusão

	de novos módulos ou remotas PROFIBUS. No entanto, valendo-se da redundância do CP e das redes PROFIBUS, foi definido um procedimento para atingir este objetivo, muito importante para sistemas que necessitam de alta disponibilidade.
Endereços IP privados para CPA e CPB	É possível conectar-se a um CP específico (CPA ou CPB) utilizando um endereço IP privado para o mesmo, para, por exemplo, obter diagnósticos específicos desse half-cluster. O Endereço de IP do CPA estará sempre associado à interface NET(i) do CPA, enquanto o Endereço de IP do CPB estará sempre associado à interface NET(i) do CPB.
IP Ativo	Nome de uma estratégia que permite que clientes Ethernet se conectem a um servidor no CP redundante, utilizando sempre o mesmo endereço de IP. Isto evita a necessidade de complexos scripts para mudar o endereço IP quando ocorrem switchovers devido à redundância. O Endereço de IP Ativo estará sempre associado à interface NET(i) do CP Ativo.
NIC Teaming	Nome de uma estratégia que permite que duas interfaces Ethernet de um half-cluster formem um par redundante compartilhando um mesmo endereço IP. Desta forma, pode-se construir redes Ethernet redundantes com maior facilidade, sem que os clientes conectados a um par NIC Teaming tenham de implementar complexos scripts para chavear endereços IP.
Redes PROFIBUS e Configuração de Falhas Vitais	O CP suporta até quatro redes PROFIBUS simples ou até duas redes PROFIBUS redundantes. Também é possível configurar se a falha de cada rede PROFIBUS é considerada vital (causa switchover) ou não.
Redes Ethernet e Configuração de Falhas Vitais	O CP suporta até 8 redes Ethernet simples ou até quatro redes Ethernet redundantes (se consideradas a interfaces frontais da NX3030). Também é possível configurar se a falha de cada rede Ethernet é considerada vital (causa switchover) ou não.
Tarefa de usuário única e cíclica	Somente uma tarefa de usuário é permitida, sendo denominada MainTask. Esta tarefa é cíclica.
POUs principais: NonSkippedPrg e ActivePrg	Na criação de um projeto redundante, o MasterTool gera automaticamente duas program POUs vazias que devem ser preenchidas pelo usuário: - NonSkippedPrg: esta program POU é executada nos dois CPs (CPA e CPB), independente do estado de redundância (Ativo ou Não-Ativo). Destina-se ao gerenciamento de diagnósticos e comandos especiais. - ActivePrg: esta program POU é executada somente no CP Ativo, e destina-se ao controle do processo do usuário final.

Tabela 6-1. Características Gerais de um CP Redundante

Dados para Compra

A configuração mínima de um CP redundante implica na aquisição dos seguintes módulos:

- Dois bastidores, que devem ser escolhidos entre quatro modelos disponíveis de acordo com os módulos que devem ser instalados:
  - NX9000: oito posições (quatro módulos duplos)
  - NX9001: doze posições (seis módulos duplos)
  - NX9002: dezesseis posições (oito módulos duplos)
  - NX9003: vinte e quatro posições (doze módulos duplos)
- Dois NX8000
- Dois NX3030
- Dois NX4010
- Dois AL-2319

Além disso, pode ser necessário adquirir os seguintes módulos opcionais:

- Um PX2612
- Um AL-2317/A

- Um AL-2317/B
- Dois módulos NX5001 para cada rede PROFIBUS simples adicional
- Quatro módulos NX5001 para cada rede PROFIBUS redundante adicional
- Dois módulos NX5000 para cada rede Ethernet simples adicional
- Quatro módulos NX5000 para cada rede Ethernet redundante adicional (NIC Teaming)

ATENÇÃO: Pode-se instalar até quatro módulos PROFIBUS em cada CP redundante. Isto significa que podemos configurar até quatro redes PROFIBUS simples ou duas redes PROFIBUS redundantes.

Princípios de Funcionamento

Nesta seção, descrevem-se as funções de um CP redundante com UCP NX3030, seu comportamento e estados. Também são apresentados conceitos e restrições de programação e configuração que serão utilizados nos próximos capítulos.

Identificação de uma UCP NX3030

Uma UCP NX3030 possui um registro de identificação, não volátil, onde é possível identificá-la como:

- Não-Redundante: não pode ser utilizada em um CP redundante (configuração padrão de fábrica).
- CPA: utilizada no CPA de um CP redundante
- CPB: utilizada no CPB de um CP redundante.

O registro de identificação pode ser ajustado com o programador MasterTool. A primeira coisa a ser feita em uma UCP NX3030, antes de carregar o projeto do CP redundante na mesma, é identificá-la como CPA ou CPB. Caso a identificação não seja realizada, muitas características da redundância não vão funcionar corretamente, como, por exemplo, a sincronização da aplicação entre os CPs.

ATENÇÃO: O registro de Identificação do CP não faz parte do projeto do CP redundante, e portanto não é salvo como parte deste projeto no computador onde o MasterTool IEC XE é executado. O registro é salvo apenas na memória não volátil da UCP.

Projeto Redundante Único

Graças ao registro de identificação descrito anteriormente, existe um único projeto para o CP redundante, idêntico para os CPA e CPB.

Parâmetros de configuração que devem ser diferentes entre CPA e CPB (ex: endereços IP de interfaces Ethernet) aparecem duplicados no projeto do CP redundante (um para CPA, outro para CPB). Cada CP irá considerar aquele que lhe corresponde, após analisar seu registro de identificação.

Estrutura do Projeto Redundante

Redundancy Template

O projeto de um CP redundante é criado automaticamente a partir de um modelo, denominado Redundancy Template.

O template parte da configuração mínima de um CP redundante, conforme definido na seção Configuração Mínima de um CP Redundante (Sem utilização do Painel PX2612). Além disso, alguns diálogos com o usuário são realizados para inserção de módulos adicionais nos half-clusters, tais como mestres PROFIBUS (NX5001) e módulos Ethernet (NX5000).

Remotas PROFIBUS devem ser inseridas pelo usuário, abaixo dos mestres PROFIBUS NX5001 já inseridos.

Além disso, são criadas tarefas e POUs básicas do tipo programa, conforme descrito nas próximas seções.

Tarefa Única e Cíclica MainTask

O projeto de um CP redundante possui uma única tarefa, denominada MainTask, que é cíclica. O usuário pode ajustar o intervalo da tarefa.

Programa MainPrg

A tarefa MainTask está associada a uma única POU do tipo programa, denominada MainPrg. O programa MainPrg é criado automaticamente.

O código do programa MainPrg é o seguinte, em linguagem ST:

```
fbRedundancyManagement();
NonSkippedPrg();
IF fbRedundancyManagement.m_fbDiagnosticsLocal.eRedState =
REDUNDANCY_STATE.ACTIVE THEN
   ActivePrg();
END_IF
```

MainPrg chama outras duas POUs do tipo programa, denominadas NonSkippedPrg e ActivePrg. NonSkippedPrg sempre é chamada, pois é executada tanto no CP Ativo quanto no Não-Ativo. Já a POU ActivePrg só é chamada quando a condição "RedDgnLoc.sGeneral.Diag.eRedState = Active" é verdadeira, ou seja, quando o CP se encontra em estado Ativo.

Portanto, o programa NonSkippedPrg será sempre executado em ambos CPs (CPA e CPB), independente do estado de redundância deste CP. Por outro lado, o programa ActivePrg será executado somente no CP que se encontra em estado Ativo.

Ao contrário do programa MainPrg, que não deve ser modificado, o usuário pode modificar os programas NonSkippedPrg e ActivePrg. Inicialmente, quando o projeto redundante é criado a partir do Redundancy Template, estes dois programa são criados "vazios", mas o usuário poderá inserir código nos mesmos.

ATENÇÃO: Quando a opção de comunicação OPC for habilitada durante a criação do projeto, o programa NonSkippedPrg não será criado vazio. Para maiores informações, consultar a seção Uso de Comunicação OPC em Projetos Redundantes.

Programa ActivePrg

O principal objetivo deste programa, que é executado somente no CP Ativo, é controlar o processo do usuário final.

Este programa normalmente atua sobre as variáveis redundantes, entre as quais encontram-se as variáveis de representação direta %I e %Q associadas ao sistema de E/S remoto. Para maiores informações, consultar o capítulo Programação de um CP Redundante - Configurações da MainTask - Programa ActivePrg.

ATENÇÃO: Havendo sucesso ou não na compilação, o MasterTool informa se a folga é respeitada e o cálculo do overhead de redundância na janela de mensagens.

Programa NonSkippedPrg

Este programa, que é executado em ambos CPs (CPA e CPB) independente do estado de redundância, é tipicamente utilizado para funções como:

- Organizar diagnósticos não-redundantes para reportar a um sistema SCADA.

- Receber e executar comandos não-redundantes a partir de um sistema SCADA.
- Gerenciar condições de switchover normalmente não contempladas automaticamente pelo CP redundante, que podem variar de usuário para usuário. Por exemplo, determinado usuário poderá executar um switchover para o CP Reserva se o CP Ativo não estiver se comunicando com o sistema SCADA, enquanto outro usuário pode não desejar um switchover nesta situação.
- Habilitar ou desabilitar I/O drivers em função do estado de redundância, por exemplo, desabilitar um mestre MODBUS RS-485 no CP Não-Ativo.
- Detectar falhas em I/O drivers em um CP Não-Ativo para evitar falhas ocultas. Alguns I/O drivers não contemplam detecção automática de tais falhas. Outros I/O drivers, como o PROFIBUS, o fazem automaticamente.
- Outras atividades que, por algum motivo, precisam ser executadas tanto no CP Ativo como no CP Reserva.

Para maiores informações, consultar o capítulo Programação de um CP Redundante - Configurações da MainTask - Programa NonSkippedPrg.

Variáveis Redundantes e Não-Redundantes

As variáveis de um CP redundante podem ser classificadas entre redundantes e não-redundantes. Variáveis redundantes são copiadas do CP Ativo para o CP Não-Ativo, no início de cada ciclo da MainTask, através dos canais de sincronismo NETA e NETB. Por outro lado, variáveis não-redundantes não são copiadas entre half-clusters e, portanto, podem ter valores diferentes nos CPs A e B.

As variáveis não-redundantes são utilizadas para armazenar informações privativas de cada half-cluster (CPA ou CPB), tais como diagnósticos de um módulo dentro deste half-cluster, incluindo os diagnósticos da redundância (estado da redundância deste half-cluster, etc...).

As variáveis redundantes dizem respeito às informações compartilhadas e relativas ao controle do processo. As variáveis associadas aos módulos de E/S são exemplos típicos de variáveis redundantes.

Variáveis %I Redundantes e Não-Redundantes

A UCP NX3030 aloca 96 Kbytes de variáveis %I (%IB0 ... %IB98303).

Os primeiros 80 Kbytes podem ser redundantes (%IB0 ... %IB81919). Os últimos 16 kbytes sempre são não-redundantes (%IB81920 ... %IB98303).

A área de 80 Kbytes que pode ser redundante tipicamente é alocada para entradas, que podem ser lidas a partir de um sistema de E/S remoto (PROFIBUS, MODBUS, etc...).

A área de 16 Kbytes não-redundante é alocada para diagnósticos privados rápidos de um half-cluster e, também, para os botões do painel de comando de redundância PX2612. Diagnósticos rápidos são aqueles que, necessariamente, devem ser atualizados a cada ciclo da MainTask.

O usuário pode configurar a quantidade de variáveis %I redundantes, entre 0 bytes e 81920 bytes, em múltiplos de 1 byte (o valor default é 16384 bytes - %IB0 ... %IB16383). A configuração adequada da quantidade de %I redundantes a partir de %IB0 é importante para reduzir o tempo necessário para sincronizar variáveis redundantes (reduzir o overhead da redundância). Por exemplo, se a aplicação real aloca apenas %IB0 ... %IB1499 para entradas redundantes, pode-se definir o tamanho da área redundante de %I em 1500 bytes.

A figura seguinte ilustra a alocação de variáveis de representação direta %I redundantes e não-redundantes, onde RI é a quantidade de %I configurados como redundantes de fato.

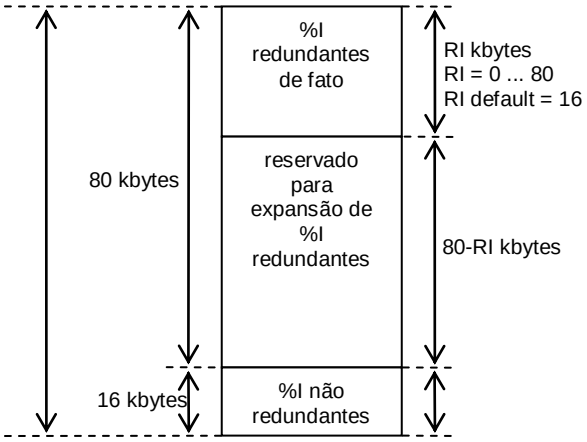

Figura 6-7. Alocação de %I Redundantes e Não-Redundantes

Variáveis %Q Redundantes e Não-Redundantes

A UCP NX3030 aloca 96 Kbytes de variáveis %Q (%QB0 ... %QB98303).

Os primeiros 80 Kbytes podem ser redundantes (%QB0 ... %QB81919). Os últimos 16 Kbytes sempre são não-redundantes (%QB81920 ... %QB98303).

A área de 80 Kbytes que pode ser redundante é dividida em duas seções:

- Os primeiros bytes são reservados para saídas que podem ser redundantes, e tipicamente alocadas para um sistema de E/S remoto (PROFIBUS, MODBUS, etc...). O tamanho dessa seção é configurável, e seu valor padrão é 16384. Ela engloba %QB0 ... %QB16383, podendo ser configurada para até 64 Kbytes (%QB0 ... %QB65535).
- Os próximos bytes são reservados para diagnósticos que podem ser redundantes, por exemplo, do sistema de E/S (diagnósticos de módulos de E/S, de interfaces de comunicação escravos PROFIBUS, etc...). Ao contrário dos diagnósticos rápidos (alocados em %I), tais diagnósticos alocados em %Q podem levar mais do que um ciclo da MainTask para serem atualizados. Por padrão, esta seção engloba 16 Kbytes (%QB65536 ... %QB81919).

A área não-redundante (%QB81920 ... %QB98303) tipicamente é alocada para diagnósticos e comandos privados de um half-cluster, e também para os LEDs e relé do painel de comando de redundância PX2612.

O usuário pode reduzir a quantidade de variáveis %Q redundantes em cada uma das duas seções que podem ser redundantes:

- Na primeira seção, o tamanho da área redundante pode ser configurado entre 0 bytes e 65535 bytes, em múltiplos de 1 byte (o valor default é 16384 bytes). A configuração adequada da quantidade de %Q redundantes nesta seção é importante para reduzir o tempo necessário para sincronizar variáveis redundantes (reduzir o overhead da redundância). Por exemplo, se a aplicação real aloca apenas %Q0 ... %Q1499 para saídas redundantes, pode-se definir o tamanho desta seção redundante de %Q em 1500 bytes.
- Na segunda seção, o tamanho da área redundante pode ser configurado entre 0 bytes e 81919 bytes, em múltiplos de 1 byte (o valor default é 16384 bytes). A configuração adequada da quantidade de %Q redundantes nesta seção é importante para reduzir o tempo necessário para sincronizar variáveis redundantes (reduzir o overhead da redundância). Por exemplo, se a aplicação real aloca apenas %QB65536 ... %QB66999 para diagnósticos redundantes, pode-se definir o tamanho desta seção redundante de %Q em 1464 bytes.

A figura seguinte ilustra a alocação de variáveis de representação direta %Q redundantes e não-redundantes, onde RQS é a quantidade de %Q de saídas configurados como redundantes na primeira seção, e RQD é a quantidade de %Q de diagnósticos configurados como redundantes na segunda seção.

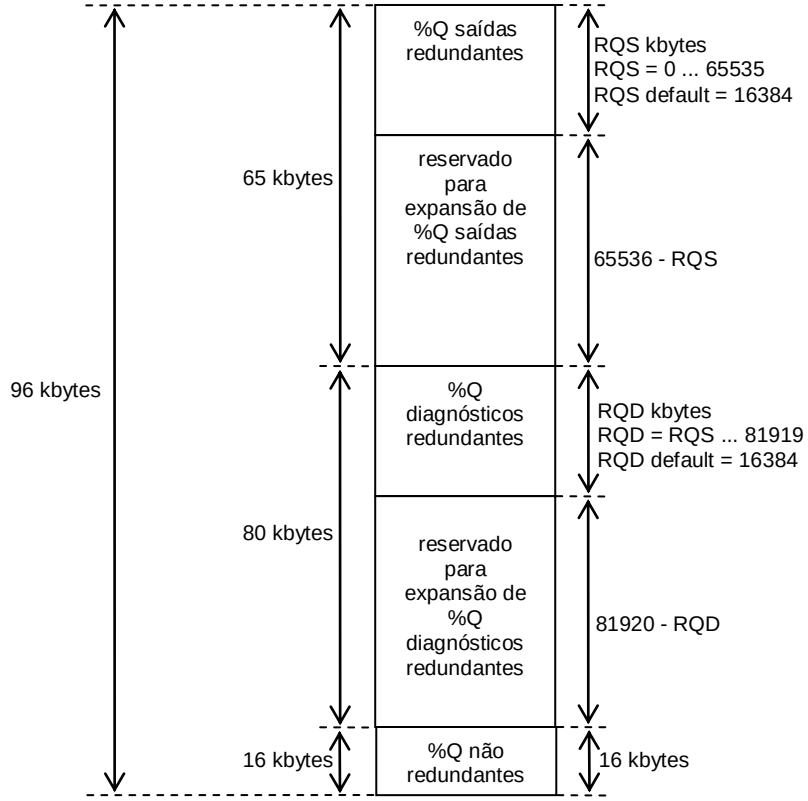

Figura 6-8. Alocação de %Q Redundantes e Não-Redundantes

Variáveis %M Redundantes e Não-Redundantes

A UCP NX3030 aloca 64 Kbytes de variáveis %M (%MB0 ... %MB65535).

Todos os 65536 bytes podem ser redundantes (%MB0000 ... %MB65535). Por padrão, a quantidade de variáveis %M redundantes é 0.

O uso de variáveis %M redundantes deve ser evitado, preferindo-se a utilização de variáveis simbólicas (ver seção Variáveis Simbólicas Redundantes e Não-Redundantes).

Variáveis Simbólicas Redundantes e Não-Redundantes

Além das variáveis de representação direta (%I, %Q e %M) que são alocadas automaticamente, o usuário pode declarar explicitamente variáveis simbólicas, dentro de POUs ou GVLs. O tamanho máximo permitido para alocação de variáveis simbólicas redundantes é de 512 Kbytes.

ATENÇÃO: Não se deve confundir variáveis simbólicas com variáveis simbólicas endereçadas através da diretiva AT. Variáveis simbólicas que utilizam a diretiva AT são nomes simbólicos atribuídos a variáveis de representação direta (%I, %Q e %M). Portanto, variáveis que utilizam a diretiva AT não alocam nenhuma memória de variáveis simbólicas.

Variáveis simbólicas são redundantes nos seguintes casos:

- Quando declaradas em POUs do tipo “programa” criadas na aplicação do usuário, com exceção do programa NonSkippedPrg
- Quando declaradas em GVLs criadas na aplicação do usuário e estas GVLs marcadas como redundantes

Variáveis simbólicas são não-redundantes nos seguintes casos:

- Quando declaradas no programa NonSkippedPrg, já descrito anteriormente na seção Programa NonSkippedPrg
- Quando declaradas em POUs do tipo "função". Observar que tais tipos de POUs normalmente deveriam alocar variáveis apenas na pilha (não estáticas), que consequentemente não precisam ser redundantes. Mesmo sabendo que o usuário pode declarar variáveis estáticas (VAR STATIC) dentro de POUs do tipo "função", isto é considerado uma má prática de programação. Tais variáveis estáticas, caso sejam criadas, serão consideradas não-redundantes
- Quando declaradas em POUs do tipo "bloco funcional". Observar que a mera declaração de um "bloco funcional" não aloca memória (o que aloca memória é instanciar um Bloco Funcional)

Deve-se observar que instâncias de blocos funcionais, declaradas dentro de POUs do tipo programa ou dentro de GVLs, comportam-se como variáveis simbólicas, ou seja, alocam memória redundante. Da mesma maneira que variáveis simbólicas, quando instâncias de bloco funcional são declaradas no programa NonSkippedPrg ou quando a GVL não é marcada como redundante, tais instâncias são não-redundantes.

Mapeamentos Múltiplos

Se o usuário desejar mapear as variáveis de comandos da redundância em mais de uma porta de comunicação (COMx ou NETx) será necessária a implementação de um controle pelo usuário na própria aplicação.

A lógica de controle a ser implementada deverá escrever nas variáveis de controle da redundância baseado nos valores das variáveis (comandos) provenientes de cada uma das portas de comunicação (COMx ou NETx). Além disso, a lógica de controle deve reinicializar as variáveis de comandos das portas de comunicação, uma vez que o controle da redundância só reinicializa suas próprias variáveis de comandos.

Segue um exemplo dessa implementação:

```
VAR
var_comando_StandBy_relacao_Ethernet : BOOL;
var_comando_StandBy_relacao_Serial : BOOL;
var_comando_Inactive_relacao_Ethernet : BOOL;
var_comando_Inactive_relacao_Serial : BOOL;
var_comando_TurnOn_relacao_Ethernet : BOOL;
var_comando_Turn_relacao_Serial : BOOL;
END_VAR

//Lógica para colocar o PLC Local para StandBy
IF var_comando_StandBy_relacao_Ethernet = TRUE THEN
   DG_NX4010.tRedundancy.RedCmdLoc.bStandbyLocal:=TRUE;
   var_comando_StandBy_relacao_Ethernet:=FALSE;
END_IF
IF var_comando_StandBy_relacao_Serial = TRUE THEN
   DG_NX4010.tRedundancy.RedCmdLoc.bStandbyLocal:=TRUE;
   var_comando_StandBy_relacao_Serial:=FALSE;
END_IF
//Lógica para colocar o PLC Local para Inativo
IF var_comando_Inactive_relacao_Ethernet = TRUE THEN
   DG_NX4010.tRedundancy.RedCmdLoc.bInactiveLocal:=TRUE;
   var_comando_Inactive_relacao_Ethernet:=FALSE;
END_IF
IF var_comando_Inactive_relacao_Serial = TRUE THEN
   DG_NX4010.tRedundancy.RedCmdLoc.bInactiveLocal:=TRUE;
   var_comando_Inactive_relacao_Serial:=FALSE;
END_IF
//Lógica para religar o PLC Local desligado pelo PX2612
IF var_comando_TurnOn_relacao_Ethernet = TRUE THEN
```

```
    DG_NX4010.tRedundancy.RedCmdLoc.bTurnOnLocal:=TRUE;
    var_comando_TurnOn_relacao_Ethernet:=FALSE;
END_IF
IF var_comando_Turn_relacao_Serial = TRUE THEN

    DG_NX4010.tRedundancy.RedCmdLoc.bTurnOnLocal:=TRUE;
    var_comando_Turn_relacao_Serial:=FALSE;
END_IF
```

Acima temos um exemplo de uma lógica em linguagem ST, onde o comando da redundância pode ser feito por duas variáveis, que provêm de portas de comunicação diferentes. No exemplo, realizamos três comandos diferentes (StandBy, Inactive e TurnOn).

Onde:

var_comando_StandBy_relacao_Ethernet: variável do tipo Bool atribuída a um Coil da comunicação Ethernet que realizará o comando para colocar o Half-Cluster Local para Stand-By.

var_comando_StandBy_relacao_Serial: variável do tipo Bool atribuída a um Coil da comunicação Serial que realizará o comando para colocar o Half-Cluster Local para Stand-By.

DG_NX4010.tRedundancy.RedCmdLoc.bStandbyLocal: este comando produz uma ação equivalente ao botão STAND-BY do PX2612, no CP local.

var_comando_Inactive_relacao_Ethernet: variável do tipo Bool atribuída a um Coil da comunicação Ethernet que realizará o comando para colocar o Half-Cluster Local para Inativo.

var_comando_Inactive_relacao_Serial: variável do tipo Bool atribuída a um Coil da comunicação Serial que realizará o comando para colocar o Half-Cluster Local para Inativo.

DG_NX4010.tRedundancy.RedCmdLoc.bInactiveLocal: este comando produz uma ação equivalente ao botão INACTIVE do PX2612, no CP local.

var_comando_TurnOn_relacao_Ethernet: variável do tipo Bool atribuída a um Coil da comunicação Ethernet que realizará o comando para religar o Half-Cluster Local depois que o relé do PX2612 o desligou.

var_comando_TurnOn_relacao_Serial: variável do tipo Bool atribuída a um Coil da comunicação Serial que realizará o comando para religar o Half-Cluster Local depois que o relé do PX2612 o desligou.

DG_NX4010.tRedundancy.RedCmdLoc.bTurnOnLocal: este comando produz uma ação equivalente ao botão STAND-BY do PX2612, no CP local.

Estruturas de Dados de Diagnósticos, Comandos e Usuário

Cada CP dispõe de diversas estruturas de dados relacionadas com a redundância. As seguintes estruturas são variáveis simbólicas que utilizam a diretiva AT para mapear variáveis do tipo %Q:

- RedDgnLoc: contém diagnósticos deste CP (local) relacionados com a redundância, como por exemplo, o estado da redundância deste CP
- RedDgnRem: é uma cópia de RedDgnLoc do outro CP, recebida via canais de sincronismo NETA / NETB. Desta forma, este CP (local) tem acesso aos diagnósticos do outro CP (remoto)
- RedCmdLoc: contém comandos que devem ser aplicados neste CP (quando têm sufixo Local) ou no outro CP (quando têm sufixo Remote). Por exemplo o campo StandbyLocal desta estrutura de dados corresponde a um comando que deve ser executado neste CP (local), enquanto o campo StandbyRemote corresponde a um comando que deve ser executado no outro CP (remoto)
- RedCmdRem: é uma cópia de RedCmdLoc do outro CP, recebida via canais de sincronismo NETA / NETB. Desta forma, este CP pode executar comandos recebidos do outro CP que tenham o sufixo Remote
- RedUsrLoc: contém 128 bytes de dados preenchidos livremente pelo usuário (ex: diagnóstico de comunicação com um sistema SCADA). Estes 128 bytes de dados podem ser intercambiados com o outro CP (remoto)

- RedUsrRem: é uma cópia de RedUsrLoc do outro CP, recebida via canais de sincronismo NETA / NETB

Na seção Manutenção, as seguintes subseções fornecerão maiores detalhes sobre estas estruturas de dados:

- Estrutura de Diagnósticos da Redundância
- Comandos da Redundância
- Informações do Usuário Trocados entre CPA e CPB

Serviços de Sincronização Cíclicos através de NETA e NETB

Esta seção descreve os três serviços de sincronização que ocorrem ciclicamente em um CP redundante, entre CPA e CPB, através dos canais de sincronismo NETA e NETB.

Estes serviços são executados no início de cada ciclo da MainTask, e na sequência em que aparecem abaixo, ou seja:

- Primeiro, executa-se o serviço Troca de Diagnósticos e Comandos
- Segundo, executa-se o serviço Sincronização de Dados Redundantes
- Terceiro, executa-se o serviço Sincronização da Lista de Forçamentos Redundantes

Troca de Diagnósticos e Comandos

Este serviço é responsável pelo intercâmbio das seguintes estruturas de dados, em cada ciclo da MainTask:

- Copiar RedDgnLoc do CPA para RedDgnRem do CPB
- Copiar RedCmdLoc do CPA para RedCmdRem do CPB
- Copiar RedUsrLoc do CPA para RedUsrRem do CPB
- Copiar RedDgnLoc do CPB para RedDgnRem do CPA
- Copiar RedCmdLoc do CPB para RedCmdRem do CPA
- Copiar RedUsrLoc do CPB para RedUsrRem do CPA

O serviço será executado utilizando apenas um dos canais de sincronismo (NETA ou NETB). Desta maneira, o serviço pode ser completado mesmo que um dos canais esteja com problemas.

Sincronização de Dados Redundantes

Este serviço é responsável pela transferência de variáveis redundantes, do CP Ativo para o CP Não-Ativo. Conforme visto anteriormente, existem variáveis redundantes simbólicas e também variáveis redundantes de representação direta (%I, %M e %Q).

Para que este serviço seja executado, diversas condições devem ser satisfeitas:

- O serviço de sincronização anterior deste ciclo da MainTask (Troca de Diagnósticos e Comando) deve ter completado com sucesso.
- Caso este CP esteja em estado Ativo, o outro deve estar em estado Não-Ativo. Por outro lado, caso este CP esteja em estado Não-Ativo, o outro deve estar em estado Ativo.
- Os projetos dos 2 CPs devem estar idênticos, exceto quando a sincronização automática de projetos estiver desabilitada (ver seção Desabilitação da Sincronização de Projetos).
- Necessário pelo menos um canal de sincronismo (NETA e/ou NETB) em estado operacional. Se ambos os canais estiverem em operação, a comunicação é distribuída entre ambos (load balance) para reduzir o tempo da sincronização. Caso apenas um canal estiver em operação, o sincronismo continuará sendo realizado somente por este canal, garantido assim a sincronização dos dados redundantes.

Sincronização da Lista de Forçamentos Redundantes

Este serviço é responsável pela transferência da lista de forçamentos redundantes do CP Ativo para o CP Não-Ativo.

Para que este serviço seja executado, diversas condições devem ser satisfeitas:

- Os dois serviços de sincronização anteriores deste ciclo (Troca de Diagnósticos e Comando, Sincronização de Dados Redundantes) devem ter completado com sucesso
- Caso este CP esteja em estado Ativo, o outro deve estar em estado Não-Ativo. Por outro lado, caso este CP esteja em estado Não-Ativo, o outro deve estar em estado Ativo
- Os projetos dos dois CPs devem estar idênticos, exceto quando a sincronização automática de projetos estiver desabilitada (ver seção Desabilitação da Sincronização de Projetos)
- Necessário pelo menos um canal de sincronismo (NETA e/ou NETB) em estado operacional. Se ambos os canais estiverem em operação, a comunicação é distribuída entre ambos (load balance) para reduzir o tempo da sincronização. Caso apenas um canal estiver em operação, o sincronismo continuará sendo realizado somente por este canal, garantido assim a sincronização dos dados redundantes

ATENÇÃO: A lista de forçamentos redundantes contém apenas forçamentos sobre variáveis redundantes. Em cada um dos CPs (CPA e CPB), pode existir uma lista diferente de forçamentos sobre variáveis não-redundantes.

Serviços de Sincronização Esporádicos através de NETA e NETB

Os seguintes serviços de sincronização são executados de forma esporádica. Ou seja, não são executados a cada ciclo da tarefa MainTask e outra tarefa do sistema executa estes serviços esporádicos em segundo plano.

Sincronização de Projetos

Este serviço é responsável por manter sincronizados os projetos dos CPs Ativo e Não-Ativo. Isto ocorre apenas quando os projetos estão diferentes e a sincronização automática de projetos está habilitada em ambos CPs.

A sincronização é sempre no sentido do CP Ativo para o Não-Ativo e basta que um dos canais de sincronismo (NETA ou NETB) esteja operacional para que este serviço seja executado.

Quando a sincronização estiver habilitada, serão sincronizados os seguintes arquivos e serviços:

- Projeto aplicativo (código executável)
- Project archive (código fonte)
- Usuários e grupos
- Direitos de acesso
- Trace

O serviço de sincronização irá iniciar em até trinta segundos, após um dos CPs entrar no modo Ativo, e, após seu início, irá checar o CRC dos projetos a cada cinco segundos.

Quando uma sincronização for iniciada, o CP Não-Ativo irá para o modo Stop, no estado de Não Configurado. Após a transferência de todos os arquivos necessários, o CP Não-Ativo irá para Run, no estado de Inicializando. Caso a transferência falhe, o CP voltará para o estado Não-Configurado.

O tempo que a sincronização levará para ser efetuada dependerá do tamanho do projeto. Em média, a taxa de transferência entre os canais de sincronismo é de aproximadamente 500 Kbytes/s.

Caso a sincronização seja interrompida (perda da comunicação entre os canais de sincronismo) durante a transferência dos arquivos do CP Ativo para o Não-Ativo, o procedimento será abortado e reinicializado quando a comunicação for restaurada. Somente após a conclusão de todo o procedimento o CP Não-Ativo irá para o modo Run.

Além de manter sincronizados os projetos, a Sincronização do Projeto também irá impedir que o CP Não-Ativo assuma estados superiores a Inicializando caso o CRC esteja diferente ou algum Online Change esteja pendente no CP Ativo.

ATENÇÃO:
Uma sincronização de projeto terá os mesmos efeitos de um download no CP Não-Ativo.
Este serviço não é executado se a sincronização automática de projetos estiver desabilitada, conforme descreve-se na seção Desabilitação da Sincronização de Projetos.
Nenhum serviço de sincronização entre o CPA e o CPB irá funcionar caso sejam invertidos os cabos dos canais de sincronismo. Por exemplo, conectar o canal NETA do CPA no NETB do CPB e o canal NETB do CPA no NETA do CPB.

ATENÇÃO:
Na atualização da versão 1.20 para versões superiores do software MasterTool IEC XE houve uma modificação no protocolo de comunicação dos canais de sincronismo. Portanto, não será possível efetuar a sincronização de dados entre dois CPs quando uma das aplicações tiver sido criada em uma versão anterior à 1.21 e outra em uma versão igual ou superior. Para voltar a efetuar o sincronismo, deve-se Não Carregar a Aplicação na Inicialização do CP com um projeto mais antigo e deixar que a aplicação seja sincronizada automaticamente, quando este CP for para o estado Não-Configurado, durante a sua inicialização.

ATENÇÃO:
Antes da versão 2.01 do MasterTool IEC XE, ao enviar o código fonte para CP Ativo, o CP Reserva passava para estado Não-Configurado para sincronização do mesmo. Contudo, ao concluir a operação de sincronização, o CP permanecia no estado Não-Configurado, sendo necessário passar o CP para o estado Reserva através do botão STAND-BY do PX2612 ou de comando equivalente. A partir da versão 2.0.1 o CP que está em Reserva irá alterar seu estado para Não-Configurado durante o processo de sincronização, mas irá voltar automaticamente quando os códigos fonte forem iguais entre os dois Half-Clusters.

Desabilitação da Sincronização de Projetos

Na seção Serviços de Sincronização Esporádicos através de NETA e NETB foram descritos serviços para a sincronização do projeto aplicativo e do project archive. Estes serviços normalmente devem estar habilitados e são úteis quando as modificações de projeto podem ser carregadas on-line no CP Ativo, sendo, a seguir, automaticamente retransmitidas para o CP Reserva via canais de sincronismo NETA / NETB.

Entretanto, existem modificações de projeto que não podem ser carregadas on-line em nenhum CP, como por exemplo a inclusão de módulos em uma remota PROFIBUS, ou a inclusão de uma nova remota PROFIBUS. Nestes casos, valendo-se da redundância do CP e da rede PROFIBUS, pode-se fazer tais modificações sem interromper o controle de processo. Um procedimento para atingir este objetivo é descrito na seção Explorando a Redundância para Carga Off-Line de Modificações sem Interrupção do Controle do Processo.

Neste procedimento, é necessário desabilitar temporariamente as sincronizações de projeto, permitindo que, por algum tempo, um dos CPs opere com a nova versão do projeto, enquanto o outro CP ainda opera com a versão antiga do projeto.

Uma UCP NX3030 possui um registro de Desabilitação da Sincronização de Projetos, não volátil, que permite desabilitar os serviços para sincronização do projeto aplicativo e do project archive. Este registro pode ser ajustado com o programador MasterTool. Basta desabilitar a sincronização de projetos em um dos dois CPs para que ela não mais ocorra.

Para desabilitar a sincronização de projetos, deve-se, primeiramente, conectar-se no CP desejado com o software MasterTool (ver seção Conexão do MasterTool com uma UCP NX3030 de um CP Redundante).

A seguir, no menu Comunicação / Configuração Básica do Cluster, abre-se a combo-box "Sincronização do Projeto", que permite selecionar uma das duas seguintes opções:

- Habilitar
- Desabilitar

Deve-se selecionar a opção "Desabilitar" e depois pressionar o botão "Escrever", ao lado desta combo-box. Uma mensagem informará se a operação teve sucesso ou não.

A configuração da desabilitação de sincronismo de projetos não faz parte do projeto redundante desenvolvido com o MasterTool. Tal configuração reside apenas em uma área de memória não-volátil da UCP, que pode ser lida ou gravada usando o MasterTool. O MasterTool não salva esta configuração em nenhum arquivo.

Esta configuração é copiada a cada ciclo da MainTask, da memória não volátil para o diagnóstico DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.bProjectSyncDisable. O usuário pode verificar este diagnóstico no CP para conferir se o comando teve sucesso, desde que o mesmo esteja em modo Run (DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag. bProjectSyncDisable deve valer 1). Caso o CP não esteja em modo Run, é possível verificar esta configuração diretamente no visor da UCP NX3030 (ver seção Diagnósticos da Redundância no Visor Gráfico da UCP NX3030).

O diagnóstico DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag. bProjectSyncDisable do CP pode ser observado também no outro half-cluster, através do diagnóstico DG_NX4010.tRedundancy.RedDgnRem.sGeneral_Diag. bProjectSyncDisable (desde que o CP Não-Ativo esteja em modo Run). Determinado CP (Ativo ou Não-Ativo) suspenderá o serviço de sincronização de projetos sempre que qualquer um dos seguintes bits estiver ligado:

- DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag. bProjectSyncDisable
  - Bit local, deste CP. Este CP está com a sincronização de projetos desabilitada
- DG_NX4010.tRedundancy.RedDgnRem.sGeneral_Diag. bProjectSyncDisable
  - Bit remoto, do outro CP. O outro CP está com a sincronização de projetos desabilitada

ATENÇÃO: O registro de Desabilitação da Sincronização de Projetos não faz parte do projeto do CP redundante, e portanto não é salvo como parte deste projeto no computador onde o MasterTool executa. O registro é salvo apenas na memória não volátil da UCP.

Configurações de Redes PROFIBUS

É possível instalar até quatro módulos Mestre PROFIBUS NX5001 em cada half-cluster. Portanto, pode-se configurar até duas redes PROFIBUS redundantes, denominadas PROFIBUS 1 e PROFIBUS 2, ou até quatro redes simples, denominadas PROFIBUS 1, PROFIBUS 2, PROFIBUS 3 e PROFIBUS 4, ou ainda uma rede redundante e duas simples, denominadas PROFIBUS 1, PROFIBUS 2 e PROFIBUS 3.

Redundância PROFIBUS

Cada uma das redes PROFIBUS pode ser redundante ou não-redundante. Por exemplo, se a rede PROFIBUS 1 for redundante, ela se dividirá em redes PROFIBUS 1 A e PROFIBUS 1 B. Caso não seja redundante, existirá apenas a rede PROFIBUS 1 A. O mesmo vale para a PROFIBUS 2.

A Figura 6-1 mostra um exemplo com uma única rede PROFIBUS (PROFIBUS 1), que é redundante (PROFIBUS 1 A e PROFIBUS 1 B).

Somente alguns tipos de remotas podem ser conectadas diretamente a este tipo de rede PROFIBUS redundante:

- PO5063V5: escravo PROFIBUS DP-V0 para remotas da Série Ponto
- PO5065: escravo PROFIBUS DP-V1 com Hart, para remotas da Série Ponto

- AL-3416: escravo PROFIBUS DP-V0 para CP AL-2004
- NX5210: escravo PROFIBUS DP-V0 para remotas da Série Nexto

A Figura 6-1 também mostra a possibilidade de conectar remotas não-redundantes a este tipo de rede PROFIBUS redundante, através do módulo AL-2433 (ProfiSwitch). Tais remotas PROFIBUS não-redundantes podem ser de qualquer marca ou modelo.

Modos de Falha PROFIBUS Vital e Não-Vital

Cada uma das redes PROFIBUS pode ser configurada em dois modos diferentes:

- Falha vital: caso esta rede falhe completamente, esta falha poderá determinar uma transição de estado de redundância no CP redundante (switchover). No caso de uma rede PROFIBUS redundante, uma falha completa consiste na falha das duas redes que a compõem.
- Falha não vital: mesmo que esta rede falhe completamente, esta falha não determinará uma transição de estado de redundância no CP redundante (switchover).

Redes Ethernet Redundantes com NIC Teaming

A Figura 6-1 mostra dois exemplos de rede Ethernet redundantes, com NIC Teaming.

No primeiro caso, a UCP NX3030 conecta-se à rede de supervisão (SCADAs), também utilizada para configuração através do MasterTool. As duas portas Ethernet da UCP NX3030 (NET 1 e NET 2) formam um par redundante NIC Teaming, interligadas em dois switches diferentes (Ethernet A e Ethernet B). Em algum ponto, estes dois switches devem ser interligados, para que exista conexão entre as duas portas NIC Teaming e disponibilidade ainda maior (contra falhas duplas).

No segundo caso, dois módulos NX5000 também formam um par redundante NIC Teaming, interligados em dois switches diferentes (Ethernet HSDN A e Ethernet HSDN B). Em algum ponto, estes dois switches devem ser interligados, para que exista conexão entre as duas portas NIC Teaming e disponibilidade ainda maior (contra falhas duplas).

Tais arquiteturas Ethernet possibilitam excelente disponibilidade, contra falhas nas portas Ethernet, em cabos e em switches.

ATENÇÃO: Caso dois módulos consecutivos, ou portas Ethernet, formem um par redundante NIC Teaming, a configuração dos parâmetros básicos e adição de dispositivos MODBUS apenas será possível na primeira interface, a seguinte, terá seus parâmetros bloqueados.

Um conjunto de duas portas Ethernet formando um par NIC Teaming possui um único endereço IP, vinculado ao par de portas. Desta forma, um cliente como um SCADA ou MasterTool, conectado a um servidor no CP, não precisa se preocupar em trocar o endereço IP caso haja falha em alguma das portas do par NIC Teaming.

Além disso, cada uma das interfaces que formam um par NIC Teaming possui diagnósticos separados, com o intuito de facilitar a depuração de falhas que eventualmente possam surgir.

Maiores detalhes sobre configuração e diagnósticos de portas NIC Teaming são fornecidos nas seções:

- Configurações das Portas Ethernet da UCP NX3030 (NET 1 e NET 2)
- Configuração dos Módulos NX5000

Métodos de Troca de IP

Um cluster redundante da Série Nexto dispõe de quatro métodos para a troca de IP das portas Ethernet dos módulos NX5000 de cada half-cluster e um método para a troca de IP das portas NET1 e NET2 da UCP NX3030. Esses métodos definem o comportamento das portas, no que se refere ao IP da mesma, de acordo com o estado atual do half-cluster (Ativo ou Não-Ativo) e com o half-cluster em questão (CPA ou CPB).

Os métodos são: IP Fixo, Troca Automática de IP, IP Ativo e Múltiplos IPs.

Ao todo, podemos relacionar até quatro IPs, de acordo com o método de troca de IP.

IP Fixo

É o método mais simples de endereçamento IP e pode ser configurado nas interfaces Ethernet dos módulos Ethernet NX5000. Nesse método, somente relacionamos os endereços de IP do CPA e do CPB. Independente do estado da redundância, CP Ativo ou Não-Ativo, o CPA sempre responderá pelo IP configurado, assim como o CPB.

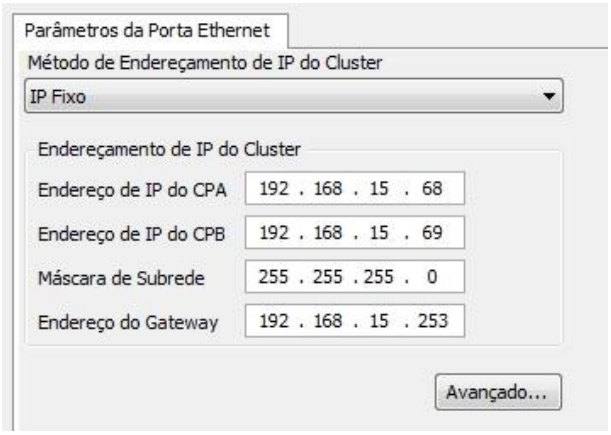
Parâmetros da Porta Ethernet
Método de Endereçamento de IP do Cluster
IP Fixo
Endereçamento de IP do Cluster
Endereço de IP do CPA
192 . 168 . 15 . 68
Endereço de IP do CPB
192 . 168 . 15 . 69
Máscara de Subrede
255 . 255 . 255 . 0
Endereço do Gateway
192 . 168 . 15 . 253
Avançado...

Figura 6-9. Método IP Fixo

Parâmetros que devem ser configurados no método de IP Fixo:

- Endereço de IP do CPA: Endereço para comunicação com o CPA
- Endereço de IP do CPB: Endereço para comunicação com o CPB
- Máscara de Subrede
- Endereço do Gateway

Troca Automática de IP

A Troca Automática de IP pode ser configurada nas interfaces Ethernet dos módulos Ethernet NX5000. Nesse método, o IP do half-cluster irá depender do estado do CP (Ativo ou Não-Ativo). A cada switchover ocorre a troca de IPs entre os half-clusters para que eles assumam o endereço IP do novo estado da redundância.

Observação: para este método de endereçamento, as portas Ethernet de ambos os CPs (CPA e CPB) irão assumir o mesmo endereço IP enquanto os dois CPs estiverem no estado não-ativo, provocando conflito de endereçamento na rede. Considerando que essa é uma situação incomum, onde nenhum CP está controlando o sistema, isso não chega a ser um problema grave, mas deve ser considerado.

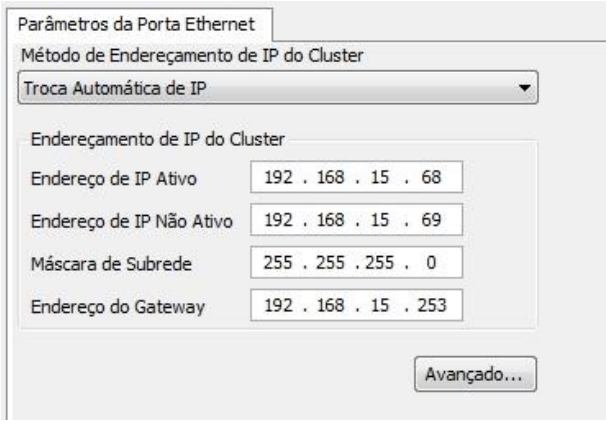
Parâmetros da Porta Ethernet
Método de Endereçamento de IP do Cluster
Troca Automática de IP
Endereçamento de IP do Cluster
Endereço de IP Ativo
192 . 168 . 15 . 68
Endereço de IP Não Ativo
192 . 168 . 15 . 69
Máscara de Subrede
255 . 255 . 255 . 0
Endereço do Gateway
192 . 168 . 15 . 253
Avançado...

Figura 6-10. Método de Troca Automática de IP

Parâmetros que devem ser configurados no método de Troca Automática de IP:

- Endereço de IP Ativo: Endereço de IP para comunicação com o CP Ativo
- Endereço de IP Não Ativo: Endereço de IP para comunicação com o CP Não-Ativo
- Máscara de Subrede
- Endereço do Gateway

IP Ativo

Esse método é o utilizado nas NETs da UCP NX3030 redundante e também é possível ser configurado nos módulos NX5000. Nesse método há um IP para o half-cluster Ativo e mais dois IPs, um para o CPA e outro para o CPB. Nas NETs da UCP NX3030 redundante, o Endereço de IP Ativo será adicionado à interface do CP Ativo, podendo ser utilizados tanto o Endereço de IP Ativo, quanto o Endereço de IP do CPX para comunicar com o CP. Já para os módulos Ethernet NX5000, o Endereço de IP Ativo irá substituir o Endereço de IP do CPX Não Ativo, quando o CP estiver em modo Ativo.

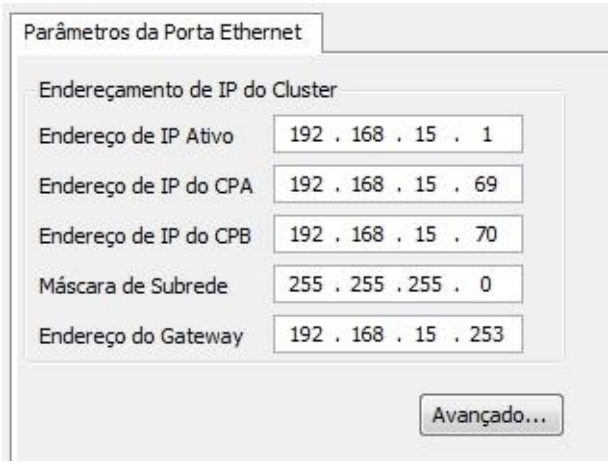
Parâmetros da Porta Ethernet
Endereçamento de IP do Cluster
Endereço de IP Ativo
192 . 168 . 15 . 1
Endereço de IP do CPA
192 . 168 . 15 . 69
Endereço de IP do CPB
192 . 168 . 15 . 70
Máscara de Subrede
255 . 255 . 255 . 0
Endereço do Gateway
192 . 168 . 15 . 253
Avançado...

Figura 6-11. Método IP Ativo – NX3030 Redundante

Parâmetros que devem ser configurados no método de IP Ativo para as NETs de uma UCP NX3030 Redundante:

- Endereço de IP Ativo: Endereço de IP adicionado à interface quando o CP estiver em estado Ativo
- Endereço de IP do CPA: Endereço para comunicação com o CPA, independente do seu estado atual
- Endereço de IP do CPB: Endereço para comunicação com o CPB, independente do seu estado atual
- Máscara de Subrede
- Endereço do Gateway

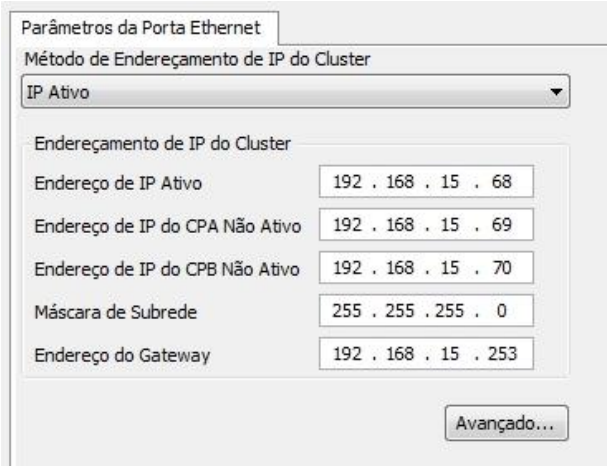
Parâmetros da Porta Ethernet
Método de Endereçamento de IP do Cluster
IP Ativo
Endereçamento de IP do Cluster
Endereço de IP Ativo
192 . 168 . 15 . 68
Endereço de IP do CPA Não Ativo
192 . 168 . 15 . 69
Endereço de IP do CPB Não Ativo
192 . 168 . 15 . 70
Máscara de Subrede
255 . 255 . 255 . 0
Endereço do Gateway
192 . 168 . 15 . 253
Avançado...

Figura 6-12. Método IP Ativo – NX5000

Parâmetros que devem ser configurados no método de IP Ativo para módulos Ethernet NX5000:

- Endereço de IP Ativo: Endereço para comunicação com o CP Ativo. Substitui o Endereço de IP do CPX Não Ativo
- Endereço de IP do CPA Não Ativo: Endereço para comunicação com o CPA, quando em estado Não-Ativo
- Endereço de IP do CPB Não Ativo: Endereço para comunicação com o CPB, quando em estado Não-Ativo
- Máscara de Subrede
- Endereço do Gateway

Múltiplos IPs

O método de Múltiplos IPs pode ser configurado nas interfaces Ethernet dos módulos Ethernet NX5000. Nesse método, há um IP para cada half-cluster e para cada estado do CP. O CPA irá assumir um endereço IP quando estiver em Ativo e outro endereço IP quando estiver em Não-Ativo. O mesmo ocorre para o CPB em função do seu estado (Ativo e Não-Ativo).

Parâmetros da Porta Ethernet
Método de Endereçamento de IP do Cluster
Múltiplos IPs
Endereçamento de IP do Cluster
Endereço de IP do CPA Ativo
192 . 168 . 15 . 68
Endereço de IP do CPA Não Ativo
192 . 168 . 15 . 69
Endereço de IP do CPB Ativo
192 . 168 . 15 . 70
Endereço de IP do CPB Não Ativo
192 . 168 . 15 . 71
Máscara de Subrede
255 . 255 . 255 . 0
Endereço do Gateway
192 . 168 . 15 . 253
Avançado...

Figura 6-13. Método Múltiplos IPs

Parâmetros que devem ser configurados no método de Múltiplos IPs:

- Endereço de IP do CPA Ativo: Endereço para comunicação com o CPA, quando o mesmo estiver em estado Ativo

- Endereço de IP do CPA Não Ativo: Endereço para comunicação com o CPB, quando o mesmo estiver em estado Não-Ativo
- Endereço de IP do CPB Ativo: Endereço para comunicação com o CPB, quando o mesmo estiver em estado Ativo
- Endereço de IP do CPB Não Ativo: Endereço para comunicação com o CPB, quando o mesmo estiver em estado Não-Ativo
- Máscara de Subrede
- Endereço do Gateway

Uso Combinado de NIC Teaming e IP Ativo

Caso determinado par de portas forme um par NIC Teaming em um CP redundante, estas portas podem implementar, ao mesmo tempo, as estratégias NIC Teaming e IP Ativo.

Por exemplo, se as portas NET 1 e NET 2 da UCP NX3030 formam um par NIC Teaming, então:

- Endereço de IP do CPA: endereço IP das portas NET 1 + NET 2 da UCP NX3030 do CPA
- Endereço de IP do CPB: endereço IP das portas NET 1 + NET 2 da UCP NX3030 do CPB
- Endereço de IP Ativo: endereço IP das portas NET 1 + NET 2 daquela UCP NX3030 que se encontra no CP Ativo

Desta maneira, associa-se a excelente disponibilidade da estratégia NIC Teaming com a praticidade da estratégia de IP Ativo, que dispensa scripts em sistemas SCADA ou em outros clientes conectados a servidores no CP Ativo.

Uso de Interfaces Ethernet com Indicação de Falha Vital

As portas Ethernet da UCP NX3030 e dos módulos NX5000 podem ser configuradas para gerar falhas vitais. Esta opção é importante para as aplicações nas quais os módulos de entradas e saídas estão distribuídos através de rede Ethernet. Neste caso, se ocorrer uma falha na porta Ethernet, isso irá gerar um switchover. Este comportamento é aplicável apenas em portas Ethernet nas quais exista pelo menos um driver de comunicação que gere falha.

Os drivers de comunicação que geram falha vital são MODBUS Client e o MODBUS Symbol Client (todas as referências a MODBUS Client nas próximas seções se aplicam aos dois casos). Já os drivers MODBUS Server, MODBUS Symbol Server e EtherCAT Master não geram falha vital. Desta forma, se uma porta Ethernet tem um driver MODBUS Client configurado e ocorre uma falha na porta Ethernet, será gerado um switchover se a opção de falha vital estiver habilitada. Se o driver configurado na porta Ethernet for um MODBUS Server, mesmo que ocorra falha na porta, isso não irá gerar uma falha vital que causa um switchover.

Para que uma falha seja considera falha vital em uma porta Ethernet de um MODBUS Client, todos os servers configurados no driver devem estar em falha. Ou seja, caso exista mais de um driver MODBUS Client configurado na mesma porta Ethernet, será considerada falha vital quando todos os servers de ambos os Clients estiverem em falha.

Quando a porta Ethernet estiver configurada para operar com NIC Teaming, a falha vital será considerada apenas quando as duas portas do par falharem.

Falha na interface Ethernet

Um switchover pode ser gerado devido a falha na interface Ethernet, por exemplo uma perda de link. A perda de link pode ser ocasionada, por exemplo, pelo rompimento de um cabo ou falha em um switch na rede Ethernet. Nestas condições, é necessário que, além de estar configurado para gerar falha vital, exista uma instância MODBUS Client configurada na interface Ethernet.

Quando o interval da MainTask for maior ou igual a 100 ms, após a falha ser detectada o switchover irá acontecer em até dois ciclos da MainTask. Quando o interval da MainTask for menor que 100 ms o switchover ocorrerá em até 100 ms mais o tempo da MainTask após a detecção da falha.

Falha em MODBUS Server conectado

O tempo para detectar a falha em uma remota MODBUS Server depende das configurações de time-out configuradas em cada MODBUS Client. Quando é detectada a falha em todos Servers, o diagnóstico bAllDevicesCommFailure (ver seção Diagnósticos MODBUS utilizados na) muda seu estado para TRUE. Quando isso acontece, o switchover irá acontecer 3 segundos após esta transição.

Uso de Comunicação OPC com Projetos Redundantes

O protocolo OPC DA pode ser configurado para comunicação de clusters redundantes com sistemas SCADA. Quando esta opção é selecionada na criação de um projeto redundante o objeto *Symbol Configuration* é adicionado ao projeto. Neste objeto são configurados as variáveis do sistema que serão enviadas para o sistema SCADA. Esta opção de comunicação é habilitada nas portas Ethernet da UCP NX3030. Para mais informações a respeito da configuração da comunicação OPC com projetos redundantes consulte a seção Configuração com CP no Servidor OPC com Redundância de Conexão deste Manual.

Estados de um CP Redundante

Em um sistema redundante, um CP (CPA ou CPB) pode assumir os seguintes estados:

- Ativo
- Reserva
- Inativo
- Não-Configurado
- Inicializando

ATENÇÃO: Frequentemente este manual utilizará a designação “Não-Ativo” para qualquer estado diferente de Ativo, ou seja, para designar qualquer um dos outros quatro estados (Reserva, Inativo, Não-Configurado e Inicializando).

A seguir, estes cinco estados são descritos brevemente. Maiores detalhes sobre os estados de um CP redundante serão abordados na seção Transições entre Estados de Redundância, ao descrever a máquina de estados e as causas das transições entre os mesmos.

Estado Não-Configurado

Este é o estado de redundância inicial. O CP encontra-se neste estado de redundância:

- Por convenção, enquanto o CP estiver desligado
- Antes de iniciar a MainTask
- Antes de chavear para o estado Inicializando
- Caso ocorra uma reinicialização através de um comando como Reset a Quente, Reset a Frio ou Reset Origem

Caso a MainTask esteja executando no estado Não-Configurado, as seguintes tarefas são executadas:

- Os mestres PROFIBUS estão desabilitados
- Os serviços de sincronização cíclicos são executados (ver seção Serviços de Sincronização Cíclicos através de NETA e NETB), desde que as condições para sua execução estejam presentes
- Os serviços de sincronização esporádicos também podem ser executados (ver seção Serviços de Sincronização Esporádicos através de NETA e NETB)

O CP ficará bloqueado no estado Não-Configurado se o outro CP estiver em estado Ativo, e o projeto deste CP for diferente do projeto do CP Ativo (exceto se a sincronização automática de projetos estiver desabilitada – ver seção Desabilitação da Sincronização de Projetos. Caso esta situação não ocorra, ocorre uma transição do estado Não-Configurado para o estado Inicializando assim que chegar uma requisição de configuração.

Algumas vezes, o CP entra no estado Não-Configurado já tendo recebido uma requisição de configuração automática, não sendo necessária uma nova requisição para mudar para o estado Inicializando. Isto ocorre, por exemplo, na energização do CP.

Outras vezes, o usuário deverá requisitar manualmente esta configuração, por exemplo, pressionando um botão do painel de comando de redundância PX2612 ou utilizando um comando na estrutura de diagnósticos. Requisições de configuração manuais tipicamente são necessárias quando alguma manutenção por parte do usuário é necessária antes de sair do estado Não-Configurado, por exemplo, se o CP atingiu o estado Não-Configurado devido a alguma falha.

Depois de sair do estado Não-Configurado, o CP pode voltar a este estado, devido a eventos tais como:

- Reinicialização (Reset a Quente, Reset a Frio ou Reset Origem)
- Desligamento do CP
- Diferença de projeto entre este CP e o CP Ativo

Estado Inicializando

Diferente de todos os outros quatro estados que podem ter duração indeterminada, o estado Inicializando é temporário, durando poucos segundos. Este estado é sempre alcançado a partir do estado Não-Configurado, através de uma requisição de configuração.

Ao entrar no estado Inicializando, diversas ações, testes e verificações são executados, para decidir qual será o próximo estado:

- Mestres PROFIBUS são habilitados em estado passivo. O modo passivo serve para testar os circuitos de transmissão e recepção PROFIBUS e o meio físico, para evitar a ocorrência de uma falha oculta
- Verificar se a identificação do CP está correta (deve ser CPA ou CPB)
- Verificar se há problemas nos parâmetros de configuração extraídos do projeto MasterTool
- Verificar a integridade do módulo NX4010
- Os serviços de sincronização cíclicos são executados (ver seção Serviços de Sincronização Cíclicos através de NETA e NETB), desde que as condições para sua execução estejam presentes
- Verificar compatibilidade de versões de firmware entre os dois CPs
- Verificar igualdade de projetos entre os dois CPs, se a sincronização automática de projetos estiver habilitada (ver seção Desabilitação da Sincronização de Projetos)
- Caso o outro CP esteja em estado Ativo, verificar a possibilidade de estabelecer comunicação PROFIBUS passiva com o mesmo. O modo passivo serve para testar os circuitos de transmissão e recepção PROFIBUS e o meio físico, para evitar a ocorrência de uma falha oculta
- Caso o outro CP esteja em estado desconhecido devido a falhas em NETA e NETB, verificar a possibilidade de estabelecer comunicação PROFIBUS passiva com o mesmo

Dependendo do resultado destas verificações e testes, o CP pode ir do estado Inicializando para qualquer um dos outros quatro estados.

Estado Ativo

Neste estado, o CP controla o processo automatizado, usando o programa ActivePrg, executado somente neste estado. O CP Ativo também atualiza o sistema de E/S remoto PROFIBUS, colocando seus mestres PROFIBUS em modo ativo. O modo ativo serve para estabelecer comunicação com as remotas (escravos) PROFIBUS.

O CP Ativo também verifica seus diagnósticos internos e requisições de switchover do usuário para determinar se um switchover é necessário. O CP normalmente só sairá do estado Ativo se souber que o outro CP está em estado Reserva, e apto para assumir como Ativo.

No entanto, existem algumas situações em que o CP Ativo poderá sair do estado Ativo mesmo sem ter certeza de que o outro CP está em estado Reserva (por exemplo, se este CP for desligado).

Estado Reserva

Neste estado, o CP está pronto para chavear para o estado Ativo, caso haja uma demanda para isso, tal como uma falha no CP Ativo.

O CP Reserva também verifica seus próprios diagnósticos, e poderá chavear para o estado Não-Configurado ou Inativo em função de algumas falhas.

Mestres PROFIBUS são habilitados em estado passivo. O modo passivo serve para testar os circuitos de transmissão e recepção PROFIBUS e o meio físico, para evitar a ocorrência de uma falha oculta. Falhas totais em redes PROFIBUS configuradas como vitais causam um chaveamento para o estado Inativo. Uma falha total em uma rede PROFIBUS atinge as duas redes que a compõem (rede PROFIBUS redundante) ou a única rede que a compõe (rede PROFIBUS não-redundante).

Se estiverem sendo usadas as interfaces ethernet com a opção de falha vital habilitada, os clientes são habilitados em estado passivo. Falhas totais em redes ethernet configuradas como vitais causam um chaveamento para o estado Inativo. Uma falha total em uma rede Ethernet atinge as duas redes que a compõem (opção de Redundância de Comunicação habilitada) ou a única rede que a compõe (opção de Redundância de Comunicação desabilitada).

Estado Inativo

Este estado normalmente é alcançado depois de alguns tipos de falhas, ou devido a alguma solicitação manual antes de uma manutenção programada.

Mestres PROFIBUS são habilitados em estado passivo. O modo passivo serve para testar os circuitos de transmissão e recepção PROFIBUS e o meio físico, para evitar a ocorrência de uma falha oculta.

Antes de deixar este estado, primeiro deve-se corrigir as falhas diagnosticadas ou executar as manutenções programadas, que causaram a transição para o estado Inativo. Depois, deve-se causar uma transição para o estado Não-Configurado já solicitando uma configuração, para logo em seguida chavear para o estado Inicializando. Depois do estado Inicializando, o CP pode:

- Retornar ao estado Inativo se determinados tipos de falhas persistem
- Retornar ao estado Não-Configurado para outros tipos de falhas
- Ir para o estado Reserva se o outro CP está no estado Ativo
- Ir para o estado Ativo se o outro CP não está no estado Ativo

Funções do Painel de Comando de Redundância PX2612

O painel de comando de redundância PX2612 é mostrado na Figura 6-4. A Figura 6-5 mostra sua serigrafia frontal com maiores detalhes e a Figura 6-6 mostra como esse painel deve ser interligado aos half-clusters CPA e CPB.

O PX2612 divide-se em duas seções, uma delas controlada pelo CPA, e outra pelo CPB. Estes controles ocorrem através dos cabos AL-2317/A para o CPA, e AL-2317/B para o CPB, e permitem que cada CP leia três botões, e escreva sobre três LEDs e sobre um relé NA.

Observando a frontal na Figura 6-5:

- CPA executa a leitura dos botões STAND-BY e INACTIVE dentro do setor PLC A
- CPA executa a leitura do botão TURN ON PLC B
- CPA executa a escrita nos três LEDs (ACTIVE, STAND-BY e INACTIVE) do setor PLC A
- CPA executa a escrita no relé RL B, utilizado para desligar o CPB
- CPB executa a leitura dos botões STAND-BY e INACTIVE dentro do setor PLC B
- CPB executa a leitura do botão TURN ON PLC A
- CPB executa a escrita nos três LEDs (ACTIVE, STAND-BY e INACTIVE) do setor PLC B
- CPB executa a escrita no relé RL A, utilizado para desligar o CPA

Botões do PX2612

Esta seção descreve as funções dos botões do PX2612.

O botão STAND-BY tem as seguintes funções:

- Solicitar um chaveamento do estado Ativo para o estado Reserva, o que pode ser útil para executar alguma manutenção programada no CP Ativo. Depois que o CP Ativo chaveia para Reserva (e consequentemente o CP Reserva chaveia para Ativo), é possível chaveá-lo de Reserva para Inativo usando o botão INACTIVE, e então executar a manutenção programada no estado Inativo
- Solicitar uma configuração que provoca um chaveamento do estado Não-Configurado para o estado Inicializando, tipicamente depois de reparar falhas que provocaram a transição para o estado Não-Configurado. Depois do estado Inicializando, normalmente espera-se que o CP vá para o estado Reserva (ou Ativo, se o outro CP não estiver no estado Ativo)
- Solicitar um chaveamento do estado Inativo para o estado Não-Configurado já solicitando uma configuração. Isto ocorre tipicamente depois de corrigir falhas que provocaram a transição para o estado Inativo. Depois do estado Não-Configurado, a configuração já deve levar ao estado Inicializando. Depois do estado Inicializando, normalmente espera-se que o CP vá para o estado Reserva (ou Ativo, se o outro CP não estiver no estado Ativo)

O botão INACTIVE solicita um chaveamento do estado Reserva para o estado Inativo, o que pode ser útil para executar alguma manutenção programada no CP Reserva. Depois desta manutenção, pode-se utilizar o botão STAND-BY para tentar voltar ao estado Reserva, passando pelos estados Não-Configurado e Inicializando (ver descrição anterior das funções do botão STAND-BY).

O botão TURN ON PLCx (x = B para CPA, ou x = A para CPB) serve para provocar um religamento do outro CP, caso este CP o tenha desligado anteriormente. Conforme será visto na seção Transições entre Estados de Redundância, existem situações excepcionais em que um CP desliga o outro ao assumir como Ativo, para evitar que haja a possibilidade de ambos CPs assumirem simultaneamente o estado Ativo.

ATENÇÃO:
Para que um pressionamento de botão seja considerado, deve-se mantê-lo pressionado no mínimo por 1 segundo. Durante esse segundo, somente esse botão deve estar pressionado (os outros 2 botões devem estar liberados).

ATENÇÃO:
Existem maneiras alternativas de gerar os mesmos efeitos dos botões STAND-BY, INACTIVE e TURN ON PLCx. Pode-se utilizar comandos gerados por este CP ou pelo CP remoto, conforme descrito preliminarmente na seção Estruturas de Dados de Diagnósticos, Comandos e Usuário. Uma descrição mais detalhada destes comandos pode ser encontrada na seção Comandos da Redundância.

LEDs do PX2612

Os LEDs do PX2612 servem para informar o estado de redundância, conforme mostra a tabela abaixo:

Estado de redundância	LED ACTIVE	LED STAND-BY	LED INACTIVE
Não-Configurado	apagado	apagado	apagado
Inicializando	aceso	aceso	aceso
Ativo	aceso	apagado	apagado
Ativo (recente)	piscando	apagado	apagado
Ativo (desligando outro CP)	aceso	piscando	apagado
Ativo (recente, e desligando outro CP)	piscando	piscando	apagado
Reserva	apagado	aceso	apagado
Inativo	apagado	apagado	aceso

Tabela 6-2. LEDs do PX2612

Cada LED pode estar apagado, aceso ou piscando. Caso esteja piscando, fica aceso por 0.5 segundos, e apagado por 0.5 segundos.

Observa-se que existem quatro animações diferentes para o estado Ativo, devido às duas seguintes características:

- Nos dois primeiros segundos em estado Ativo o LED ACTIVE pisca, e depois permanece aceso. Esta animação foi criada porque, nestes primeiros instantes em estado Ativo, normalmente o CP não aceitará comandos que o façam sair do estado Ativo. Maiores detalhes sobre este comportamento do CP Ativo serão fornecidos na seção Transições entre Estados de Redundância e na seção Primeiros Instantes em Estado Ativo
- Caso este CP esteja desligando o outro CP através do seu relé do PX2612, o LED STAND-BY pisca, e do contrário permanece apagado

Relés do PX2612

O PX2612 possui dois relés NA. O CPA pode controlar RL B, para comandar o desligamento do CPB. O CPB pode controlar RL A, para comandar o desligamento do CPA.

Tais situações de desligamento ocorrem em situações excepcionais, descritas na seção Transições entre Estados de Redundância.

Transições entre Estados de Redundância

A figura seguinte mostra a máquina de estados da redundância, ilustrando todas as possíveis transições entre estados de redundância.

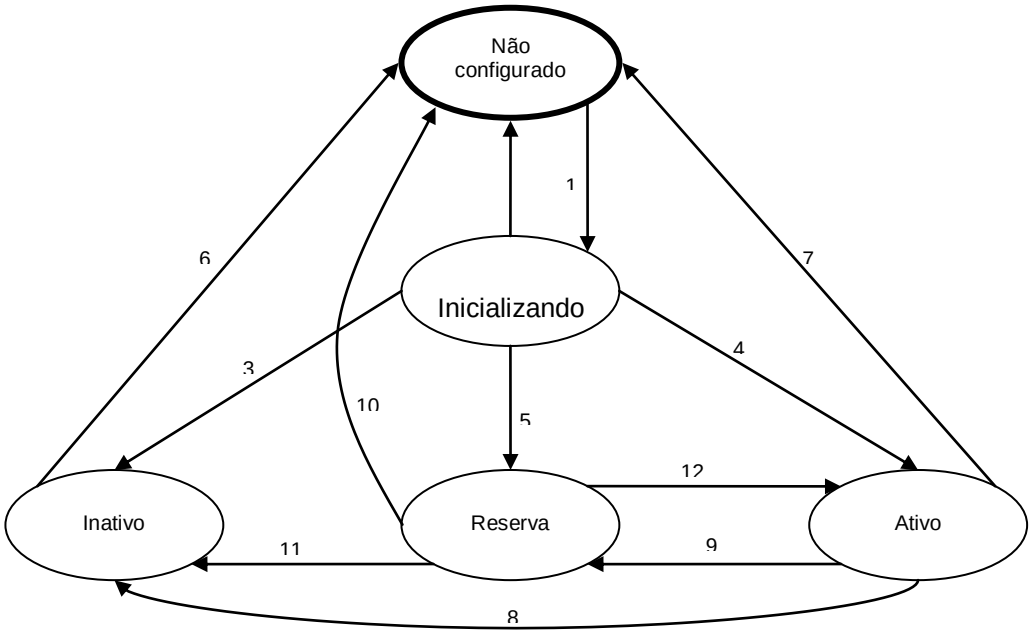

Figura 6-14. Máquina de Estados da Redundância

As seguintes subseções descreverão todas estas transições, e as causas que podem dispará-las. Para interpretar corretamente o funcionamento desta máquina de estados, é preciso estabelecer algumas regras e sequências:

- Transições que se originam do mesmo estado devem ser avaliadas na sequência dada pelo número da transição. Por exemplo, as transições 2, 3, 4 e 5 se originam do estado Inicializando. Neste exemplo, avalia-se primeiro a transição 2, depois a 3, depois a 4 e finalmente a 5. Caso a transição 2 seja disparada, as transições 3, 4 e 5 nem serão avaliadas
- Dentro de uma subseção específica que descreve uma transição, diversas condições podem disparar esta transição. Estas condições devem ser avaliadas na ordem que aparecem na

subseção. Qualquer uma destas condições que se tornar verdadeira pode causar a transição. Se uma condição causar a transição, as próximas condições não serão avaliadas

- Transições só podem ser disparadas se o CP estiver energizado e a MainTask estiver executando. Do contrário, assume-se que o CP esteja no estado Não-Configurado
- Em diversos casos, mencionam-se transições causadas por botões do painel PX2612. Deve-se lembrar que existem alternativas para estes botões, que são comandos internos provenientes deste CP ou do outro CP (via NETA / NETB). Tais comandos foram mencionados preliminarmente na seção Estruturas de Dados de Diagnósticos, Comandos e Usuário, e serão melhor detalhados na seção Comandos da Redundância. Nas subseções seguintes, por simplicidade, estes comandos alternativos não serão citados, mas deve-se ter consciência que eles podem causar as mesmas transições do botão equivalente no PX2612

Transição 1 - Não-Configurado para Inicializando

ATENÇÃO: As condições desta subseção não devem ser avaliadas caso o outro CP esteja em estado Ativo e os projetos estejam diferentes. Este CP deve permanecer no estado Não-Configurado enquanto seu projeto estiver diferente do projeto do outro CP, se o outro CP estiver em estado Ativo. Esta nota não é válida se a sincronização automática de projetos estiver desabilitada (ver seção Desabilitação da Sincronização de Projetos), pois neste caso admite-se diferenças de projetos entre os CPs.

- Um pedido de configuração já existia ao entrar no estado Não-Configurado. Isto ocorre na energização do CP, e também em outras situações, descritas nas próximas subseções
- O botão STAND-BY foi pressionado ou o comando de ir para Reserva foi executado nos diagnósticos da redundância durante o estado Não-Configurado. Isto causa um pedido de configuração manual. O usuário tipicamente aperta STAND-BY ou executa o comando para ir para Reserva depois de reparar falhas que anteriormente levaram este CP ao estado Não-Configurado

Transição 2 - Inicializando para Não-Configurado

- Este CP foi desligado ou reinicializado (Reset a Quente, Reset a Frio ou Reset Origem) ou sua UCP entrou em modo Stop
- O registro de identificação deste CP é inválido (diferente de CPA ou CPB)
- Existem erros lógicos de configuração no projeto recebido do MasterTool IEC XE
- O outro CP está no estado Ativo, e a versão de firmware deste CP é incompatível com a versão de firmware do CP Ativo
- O outro CP está em estado Ativo, e o projeto deste CP é diferente do projeto do CP Ativo. Além de ir para o estado Não-Configurado, uma solicitação de configuração é feita. Desta maneira, depois que os projetos forem sincronizados, o CP sai automaticamente do estado Não-Configurado para o estado Inicializando. Esta condição não é avaliada se a sincronização automática de projetos estiver desabilitada (ver seção Desabilitação da Sincronização de Projetos)

Transição 3 - Inicializando para Inativo

- Módulo NX4010 não detectado no barramento, ou falha no seu microprocessador
- Algum dos canais de sincronismo (NETA ou NETB) está em falha, e este CP sabe que esta falha foi causada por componentes de hardware ou software internos (falhas internas de NETA ou NETB)
- O outro CP está em estado Ativo. No entanto, não é possível sincronizar os dados redundantes ou a lista de forçamentos redundantes
- O estado do outro CP não pode ser descoberto via NETA / NETB, mas este CP consegue monitorar tráfego em alguma das redes PROFIBUS configurada em modo de falha vital. Desta maneira, parece que o outro CP está controlando o processo, embora NETA / NETB não estejam funcionando para ter certeza disso

- Houve perda de link em uma interface Ethernet configurada como falha vital.

Transição 4 - Inicializando para Ativo

- O outro CP encontra-se em estado Não-Ativo. Antes de fazer esta transição, esta condição deve permanecer verdadeira durante algum tempo. Quando o CPA e CPB são energizados simultaneamente, o CP que terminar a inicialização do sistema antes assume como Ativo
- O estado do outro CP não pode ser descoberto via NETA / NETB, e além disso este CP não consegue monitorar tráfego em nenhuma das redes PROFIBUS configuradas em modo de falha vital, ou então não possui nenhuma rede PROFIBUS configurada em modo de falha vital. Portanto, parece realmente que o outro CP está desligado ou fora de execução. Quando utilizando projeto com painel PX2612, além de chavear para Ativo, este CP desligará o outro CP utilizando seu relé do PX2612. Esta condição deve se manter durante algum tempo antes de fazer esta transição

Transição 5 - Inicializando para Reserva

- O outro CP encontra-se em estado Ativo e os serviços de sincronização de dados redundantes e de sincronização da lista de forçamentos redundantes estão funcionando corretamente

Transição 6 - Inativo para Não-Configurado

- Este CP foi desligado ou reinicializado (Reset a Quente, Reset a Frio ou Reset Origem) ou sua UCP entrou em Modo Stop
- O botão STAND-BY foi pressionado no PX2612 ou o comando de ir para Reserva foi executado nos diagnósticos da redundância. Além de ir para o estado Não-Configurado, uma solicitação de configuração é feita. Desta maneira, o CP sai automaticamente do estado Não-Configurado para o estado Inicializando. O usuário tipicamente aperta este botão depois de reparar uma falha que levou o CP anteriormente ao estado Inativo
- Este CP está com a sincronização desabilitada e o projeto está diferente do CP Ativo, ao pressionar o botão STAND-BY, o CP vai de inativo para Não-Configurado

Transição 7 - Ativo para Não-Configurado

- Este CP foi desligado ou reinicializado (Reset a Quente, Reset a Frio ou Reset Origem) ou sua UCP entrou em Modo Stop

Transição 8 - Ativo para Inativo

- Módulo NX4010 não detectado no barramento, ou falha no seu microprocessador. Além disso, este CP sabe que o outro CP estava em estado Reserva antes desta falha ocorrer. Esta condição não é avaliada nos primeiros 2 segundos em estado Ativo
- Este CP perdeu comunicação com o outro CP via NETA e NETB devido a uma falha interna, mas sabe que o outro CP estava em estado Reserva pouco antes desta falha ocorrer. Esta condição não é avaliada nos primeiros 2 segundos em estado Ativo
- Este CP não consegue controlar todas as redes PROFIBUS configuradas em modo de falha vital, e sabe que o outro CP está em estado Reserva. Esta condição não é avaliada nos primeiros 2 segundos em estado Ativo
- Este CP detectou falha total nas redes Ethernet configuradas em modo de falha vital, e sabe que o outro CP está em estado Reserva

Transição 9 - Ativo para Reserva

- Os dois CPs, por algum motivo, estão no estado Ativo, e este conflito deve ser resolvido. O CPA chaveará para o estado Reserva caso este conflito dure certo tempo, e o CPB fará o mesmo após um tempo diferente, que é menor do que o tempo do CPA. Desta maneira, em caso de conflito, o CPA tem prioridade para continuar em estado Ativo. Caso o estado do outro CP seja

desconhecido o CP Ativo não irá para o estado Reserva. Isso acontece no caso do projeto redundante não utilizar painel PX2612

- O botão STAND-BY foi pressionado, ou o comando de ir para Reserva foi executado nos diagnósticos da redundância, e este CP sabe que o outro CP encontra-se no estado Reserva. Esta condição não é avaliada nos primeiros 2 segundos em estado Ativo

Transição 10 – Reserva para Não-Configurado

- Este CP foi desligado ou reinicializado (Reset a Quente, Reset a Frio ou Reset Origem) ou sua UCP entrou em Modo Stop
- O outro CP encontra-se em estado Ativo e o projeto deste CP é diferente do projeto do CP Ativo. Além de ir para o estado Não-Configurado, uma solicitação de configuração é feita. Desta maneira, depois que os projetos forem sincronizados, o CP sai automaticamente do estado Não-Configurado para o estado Inicializando. Esta condição não é avaliada se a sincronização automática de projetos estiver desabilitada (ver seção Desabilitação da Sincronização de Projetos)
- O outro CP está no estado Ativo, e a versão de firmware deste CP é incompatível com a versão de firmware do CP Ativo

Transição 11 – Reserva para Inativo

- Módulo NX4010 não detectado no barramento, ou falha no seu microprocessador
- O botão INACTIVE foi pressionado no PX2612 ou o comando de ir para Inativo foi executado nos diagnósticos da redundância. Isto é feito tipicamente para executar uma manutenção programada no CP Não-Ativo. Deve-se evitar fazer manutenções programadas no CP Reserva, por isso, antes é aconselhável chaveá-lo para Inativo
- O outro CP encontra-se em estado Ativo. No entanto, o serviço de sincronização de dados redundantes não funcionou corretamente nos últimos quatro ciclos da MainTask, ou o serviço de sincronização de diagnósticos redundantes não funcionou corretamente nos últimos dois ciclos da mesma
- O outro CP encontra-se em estado Ativo. No entanto, este CP não consegue monitorar tráfego PROFIBUS em todas as redes configuradas em modo de falha vital
- O outro CP encontra-se em estado Ativo. No entanto, este CP detectou falha nas portas Ethernet configuradas em modo de falha vital

Transição 12 – Reserva para Ativo

- O estado do outro CP é desconhecido devido a falhas em NETA e NETB. Neste caso, além de ir para o estado Ativo, por segurança este CP desliga o outro CP usando o relé do PX2612. Quando a Redundância não usa painel PX2612 e não existe rede PROFIBUS DP esta condição não é gerada permanecendo o CP em estado Reserva. Nesta condição caso a falha tenha sido gerado pelo outro CP, para voltar a controlar o processo é necessário executar o comando para passar o CP para o estado Inativo e depois o comando para passar o CP para o estado Reserva. Quando esta sequencia for executada este CP irá assumir o estado Ativo
- O estado do outro CP é conhecido e diferente de Ativo

Primeiros Instantes em Estado Ativo

Caso a aplicação utilize o Painel PX2612, nos primeiros 2 segundos em estado Ativo, conforme já foi descrito na seção Funções do Painel de Comando de Redundância PX2612, o LED ACTIVE pisca, e só depois deste tempo fica aceso. Nos casos em que não é utilizado painel PX2612 este comportamento pode ser observado nos diagnósticos da redundância, na variável que reflete o estado do LED.

Nesta condição, diversas transições que normalmente poderiam tirar o CP do estado Ativo não são avaliadas (ver subseções anteriores que definem transições a partir do estado Ativo). Por exemplo, durante este tempo, não adianta apertar o botão STAND-BY ou o comando de ir para Reserva nos diagnósticos da redundância para tentar fazer o CP ir para o estado Reserva.

Apenas duas condições permitem que o CP saia do estado Ativo deste estado transitório. Estas condições são as seguintes:

- Este CP foi desligado ou reinicializado (Reset a Quente, Reset a Frio ou Reset Origem) ou sua UCP entrou em Modo Stop, causando transição para Não-Configurado
- Os dois CPs, por algum motivo, estão no estado Ativo, e este conflito deve ser resolvido. O CPA chaveará para o estado Reserva caso este conflito dure certo tempo, e o CPB fará o mesmo após um tempo diferente, que é menor do que o tempo do CPA. Desta maneira, em caso de conflito, o CPA tem prioridade para continuar em estado Ativo

Além disso, nos primeiros instantes que um CP assume o estado Ativo, alguns diagnósticos que não são redundantes podem não ser válidos, como, por exemplo, os diagnósticos dos módulos NX5000 e NX5001. O método para não considerar estes diagnósticos possivelmente "inválidos" é descrito na seção Leitura de Diagnósticos Não-Redundantes.

Falhas mais Comuns Causadoras de Switchovers Automáticos entre Half-Clusters

Nesta seção, são listadas as falhas mais comuns que, de forma automática, causam um switchover do CP Ativo para Não-Ativo, e do CP Reserva para Ativo. Estas falhas disparam um sub-conjunto daquelas transições examinadas na seção Transições entre Estados de Redundância.

- Falta de alimentação no CP Ativo. É importante que os dois CPs tenham sistemas de alimentação redundantes, para que uma falha comum na alimentação não afete também o CP Reserva
- Falha na fonte de alimentação NX8000 do CP Ativo
- Falha no barramento do bastidor (NX9001, NX9002 ou NX9003) do CP Ativo
- Falhas na UCP NX3030 do CP Ativo, tais como:
    - Cão-de-guarda
    - Reinicialização (Reset a Quente, Reset a Frio ou Reset Origem)
    - Parada (Stop)
    - Falha nas interfaces de barramento em um ou ambos canais de sincronismo NETA e NETB
- Falhas no NX4010 do CP Ativo, tais como:
    - Módulo não reconhecido no barramento pela UCP NX3030
    - Falha no microprocessador do NX4010, que impedem atualização dos diagnósticos internos de NETA / NETB e do painel de controle PX2612 (botões, LEDs e relé)
    - Falha internas que afetam um ou ambos canais de sincronismo NETA e NETB
- Falha total de uma rede PROFIBUS no CP Ativo, caso esta rede esteja configurada em modo vital. Caso a rede PROFIBUS seja redundante, ambas redes que a compõem devem estar em falha (falha dupla)
- Falha total de uma rede Ethernet no CP Ativo, caso esta rede esteja configurada com vital. Caso a rede Ethernet seja redundante , ambas redes que a compõem devem estar em falha (falha dupla)

Falhas Associadas a Switchovers entre Half-Clusters Gerenciados pelo Usuário

Entre as transições examinadas na seção Transições entre Estados de Redundância, algumas possibilitam que o usuário gerencie switchovers entre half-clusters, devido a falhas que normalmente não geram switchovers de forma automática.

Existem casos muito particulares, que dependem da filosofia de cada cliente. Considere-se um exemplo em que o sistema SCADA perde comunicação com o CP Ativo, mas mantém-se comunicando com o CP Reserva.

Alguns clientes poderão preferir um switchover manual, onde o operador pressiona o botão STAND-BY do PX2612, para o CP Ativo. O switchover provoca a retomada da comunicação com o novo CP Ativo.

Uma solução alternativa seria provocar o switchover enviando um comando do SCADA para o CP Reserva, que o repassaria para o CP Ativo via NETA / NETB, usando as estruturas de dados

RedCmdLocal (CP Reserva) e RedCmdRem (CP Ativo) para transportar um comando equivalente ao botão STAND-BY do PX2612.

Também seria possível o próprio CP Ativo detectar sua perda de comunicação com o SCADA, e ativar um comando em RedCmdLocal, equivalente ao botão STAND-BY do PX2612. Esta seria uma solução totalmente automática e sem intervenção do operador, que tipicamente seria implementada na POU ActivePrg.

Através de estruturas de dados como aquelas citadas na seção Estruturas de Dados de Diagnósticos, Comandos e Usuário, é possível trocar diagnósticos e comandos entre os half-clusters via NETA e NETB e, desta maneira, o usuário pode executar gerenciamentos especiais de redundância, para falhas que normalmente não causariam switchovers. Maiores detalhes sobre estas estruturas de dados serão fornecidos nas seções:

- Estrutura de Diagnósticos da Redundância
- Comandos da Redundância
- Informações do Usuário Trocados entre CPA e CPB

Abaixo é exemplificado como o usuário pode gerenciar falhas e executar um switchover devido a um erro nas interfaces Ethernet do CP Ativo (o código deve ser utilizado na POU ActivePrg):

```
//Verifica se NIC Teaming está habilitado ou não.
IF ((DG_NX3030.tDetailed.Ethernet.NET1.szIP = '0.0.0.0') OR
(DG_NX3030.tDetailed.Ethernet.NET2.szIP = '0.0.0.0')) THEN
   //NIC Teaming habilitado: erro nas duas NETs para executar switchover.
   IF (DG_NX3030.tDetailed.Ethernet.NET1.bLinkDown AND
   DG_NX3030.tDetailed.Ethernet.NET2.bLinkDown) THEN
      //Coloca o CP Local em StandBy.
      DG_NX4010.tRedundancy.RedCmdLoc.bStandbyLocal := TRUE;
   END_IF
ELSE
   //NIC Teaming desabilitado: erro em uma NET para executar switchover.
   IF (DG_NX3030.tDetailed.Ethernet.NET1.bLinkDown OR
   DG_NX3030.tDetailed.Ethernet.NET2.bLinkDown) THEN
      //Coloca o CP Local em StandBy.
      DG_NX4010.tRedundancy.RedCmdLoc.bStandbyLocal := TRUE;
   END_IF
END_IF
```

ATENÇÃO:
Quando duas interfaces formarem um par NIC Teaming, a interface inativa sempre terá o endereço de IP 0.0.0.0. Este não é um endereço de IP válido e não é possível configurar manualmente uma interface com este endereço.

Tolerância a Falhas

O objetivo principal de um CP redundante é o aumento da disponibilidade do sistema. A disponibilidade é a razão entre o tempo em que o sistema está funcionando corretamente e o tempo total desde a implantação do sistema. Por exemplo, se um sistema foi implantado há 10 anos, e durante este período esteve parado no tempo total de um ano devido a falhas, então sua disponibilidade foi de apenas 90%. Disponibilidades desta ordem são geralmente inaceitáveis para sistemas críticos, sendo que valores da ordem de 99,99% ou ainda superiores podem ser solicitados nestes sistemas.

Para atingir disponibilidades desta ordem, são necessárias diversas estratégias:

- Utilização de componentes mais confiáveis (com alto MTBF, ou tempo médio entre falhas), o que contribuirá para o aumento do MTBF do sistema como um todo
- Utilização de redundância pelo menos para os componentes mais críticos ou componentes com menor MTBF, de tal maneira que a falha de um componente possa ser tolerada sem parar o

sistema. Se a redundância for implementada através da duplicação de componentes, será necessário que os dois falhem para que o sistema como um todo fique indisponível

- Alta cobertura de diagnósticos, em especial de componentes redundantes. A redundância de componentes é pouco útil para o aumento da disponibilidade quando não se pode descobrir que um componente redundante falhou. Neste caso, a primeira falha em um dos componentes ainda não compromete o sistema, mas por permanecer oculta, algum dia acontecerá a segunda falha que comprometerá o sistema, já que a primeira falha ainda não foi reparada. As falhas podem ser classificadas entre diagnosticáveis e ocultas, sendo altamente desejável que todas as falhas de componentes redundantes sejam diagnosticáveis
- Também é importante que componentes não-redundantes tenham ampla cobertura de diagnósticos, pois muitas vezes o sistema pode continuar funcionando mesmo com a falha de um componente não-redundante. O componente pode não estar sendo solicitado. Por exemplo, um relé com contato aberto, que raramente tem sua bobina acionada, não tem sua falha detectada até o momento em que o sistema solicitar seu fechamento
- Baixo tempo de reparo de componentes não-redundantes. A falha de um componente não-redundante pode comprometer o sistema, e durante o reparo, o sistema estará indisponível
- Possibilidade de reparar ou substituir um componente redundante sem parar o sistema. Se esta possibilidade existe, obtém-se um grande aumento de disponibilidade. Do contrário, deve-se programar uma parada do sistema para substituir o componente, e o tempo de reparo conta como tempo indisponível
- Baixo tempo de reparo de componentes redundantes. A falha de um componente redundante não compromete o sistema, mas durante seu reparo, eventualmente pode ocorrer uma falha em seu par redundante. Por este motivo, é importante que a falha seja reparada brevemente, após diagnosticada. Quanto maior o tempo de reparo, maior a probabilidade de acontecer uma segunda falha no componente redundante durante o reparo da primeira falha, o que comprometeria o sistema. Portanto, quanto maior o tempo de reparo, menor a disponibilidade do sistema
- Programar testes off-line periódicos em componentes para detectar falhas não diagnosticáveis automaticamente pelo sistema. O objetivo é detectar falhas ocultas, especialmente em componentes redundantes ou componentes simples que não estejam sendo solicitados (por exemplo, um relé de segurança). Testes off-line às vezes implicam em paradas no sistema, o que diminui a disponibilidade. Normalmente aproveita-se ocasiões especiais, tais como paradas programadas de manutenção da planta. Quanto maior o período entre testes off-line, maior o tempo em que uma falha pode permanecer oculta, e portanto maior a probabilidade de uma falha comprometer o sistema, ou seja, menor a disponibilidade do sistema

Estes princípios foram considerados no projeto de CPs redundantes com UCP NX3030.

As próximas subseções analisam diversos tipos de falhas e como são toleradas ou não toleradas, e se existem switchovers associados às falhas toleradas.

Falhas Simples com Indisponibilidade

Alguns componentes, por não serem duplicados, não toleram nem sequer falha simples sem causar algum tipo de indisponibilidade. Em um CP redundante com UCP NX3030, trata-se dos seguintes componentes:

- Remotas (escravos) PROFIBUS em uma rede PROFIBUS não-redundante
- Remotas (escravos) Ethernet em uma rede Ethernet não redundante
- Módulos de E/S

A intolerância a falhas de uma rede PROFIBUS não-redundante pode ser resolvida optando-se por uma rede PROFIBUS redundante, que é aconselhada em sistemas que demandam alta tolerância a falhas. A Figura 6-1 mostra um exemplo de arquitetura com uma rede PROFIBUS redundante. Da mesma forma a intolerância a falhas de uma rede Ethernet não redundante pode ser resolvida usando-se uma rede Ethernet redundante com a configuração de NIC Teaming.

Quanto à indisponibilidade de um módulo de E/S, deve-se observar que a mesma não se constitui em uma indisponibilidade total do sistema. Ela se constitui em uma indisponibilidade parcial, somente das malhas de controle que utilizam este módulo de E/S.

Embora não haja previsão de redundância de módulos de E/S, a aplicação do usuário pode gerenciá-la em casos especiais. Por exemplo, o usuário pode inserir três módulos de entradas analógicas em três remotas PROFIBUS diferentes, e implementar um esquema de votação entre triplas de entradas analógicas, para algum sistema crítico. No entanto, tais soluções, como foi enfatizado, devem ser gerenciadas pelo usuário. Não existe nenhum suporte automático para as mesmas. Tais soluções, em geral, também implicam na redundância de transdutores e atuadores no campo.

Falhas Simples sem Indisponibilidade Causando um Switchover

Alguns componentes redundantes toleram falhas simples sem causar indisponibilidade, mas causam switchover:

- Bastidores (NX9000, NX9001, NX9002 ou NX9003)
- Fontes de alimentação (NX8000)
- UCPs (NX3030)
- Módulo NX4010
- Módulos NX5001 (mestres PROFIBUS) em configuração de rede PROFIBUS não-redundante
- Módulos NX5000 (Ethernet) em configurações sem NIC Teaming
- Interface escravo PROFIBUS de uma remota redundante (PO5063V5, PO5065, NX5210 ou AL-3416). Neste caso, diferente dos anteriores, o switchover acontece dentro da remota, entre as redes PROFIBUS A e B.

ATENÇÃO: No caso de falha da UCP NX3030 ou do módulo NX4010 em arquiteturas onde não é utilizado painel PX2612 nem rede PROFIBUS, o CP irá permanecer no estado atual. Nesta caso, se a falha acontece no half-cluster Ativo, ocorre indisponibilidade do sistema.

Falhas Duplas sem Indisponibilidade Causando um Switchover

Alguns componentes são duplicados em cada half-cluster, desta maneira, antes de causar um switchover, é necessário que ambos falhem:

- Módulos NX5001 (mestres PROFIBUS) em configuração redundante, configurados em modo de falha vital.
- Módulos NX5000 (Ethernet) em configurações com NIC Teaming (redundância gerenciada pelo usuário).

Overhead da Redundância

Uma aplicação redundante causa um aumento no tempo de processamento de uma aplicação, quando comparado ao tempo necessário para uma aplicação equivalente não-redundante.

Este tempo adicional deve-se principalmente à execução dos serviços de sincronização cíclicos, descritos na seção Serviços de Sincronização Cíclicos através de NETA e NETB, além de um tempo menor para o próprio gerenciamento da redundância (máquina de estados, etc...). O tempo adicional total, devido à redundância (overhead da redundância), é estimado pelo MasterTool e exibido na janela de Mensagens, após compilar o projeto do CP redundante.

ATENÇÃO: O overhead calculado pelo MasterTool considera uma lista de forçamentos de variáveis redundantes vazia.

Cabe ao usuário definir um intervalo para a MainTask que acomode:

- O tempo adicional para redundância estimado pelo MasterTool

- O tempo necessário para executar as POUs principais (NonSkippedPrg e ActivePrg). Este tempo normalmente é medido após o desenvolvimento do projeto (descontando o tempo adicional para redundância)
- Alguma folga do ciclo de MainTask, para execução de outras tarefas da UCP (sistema operacional, I/O drivers PROFIBUS, MODBUS, etc...). O percentual desta folga pode variar de acordo com o desempenho solicitado destas outras tarefas. Por exemplo, se a comunicação MODBUS com o sistema SCADA precisa alocar muito processamento para atingir um desempenho satisfatório, esta folga deverá ser aumentada

ATENÇÃO: Dependendo do alinhamento de memória, o número de bytes utilizados no cálculo do overhead de redundância poderá ser maior do que o total de bytes declarados nas variáveis.

Programação de um CP Redundante

Wizard para Criação de um Novo Projeto Redundante

Para criar um novo projeto redundante, deve-se utilizar o comando Arquivo / Novo Projeto, e selecionar o Projeto MasterTool Padrão.

Inicialmente, o usuário deve informar o nome que deseja dar ao projeto e o diretório onde deseja armazená-lo, conforme mostra a Figura 6-15.

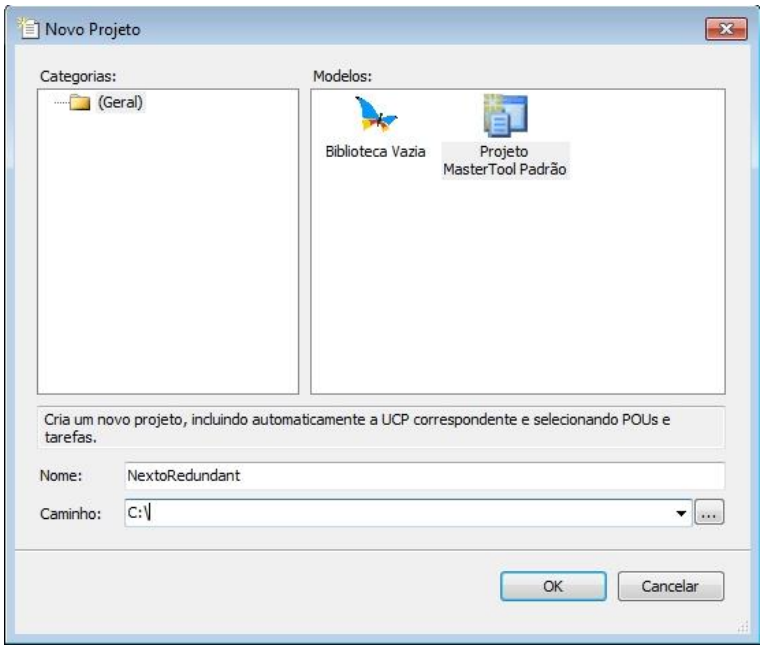

Figura 6-15. Novo Projeto

A seguir, o Wizard que gera o projeto de redundância realiza alguns questionamentos ao usuário quanto à configuração desejada, os quais devem ser respondidos sucessivamente.

O primeiro ponto a ser definido é a configuração inicial de hardware do half-cluster:

- Selecionar o modelo da UCP: Como a redundância está implementada somente na NX3030, a mesma deve ser selecionada pelo usuário
- Selecionar o modelo do rack: Existem três opções de bastidores disponíveis e a escolha do mesmo dependerá da quantidade de módulos utilizados na redundância. O MasterTool consiste o tamanho do rack de acordo com a quantidade de redes configuradas (próximo item do Wizard)
- Selecionar o modelo da fonte de alimentação
- Selecionar a configuração da redundância. Para um projeto redundante é necessário escolher a opção Com Redundância

- Selecionar o modo de operação da redundância. Neste caso as opções são operação com painel de redundância (PX2612) ou sem ele
- Selecionar se a opção de comunicação OPC estará habilitada ou não
- Selecionar se será usada redundância na expansão de barramento

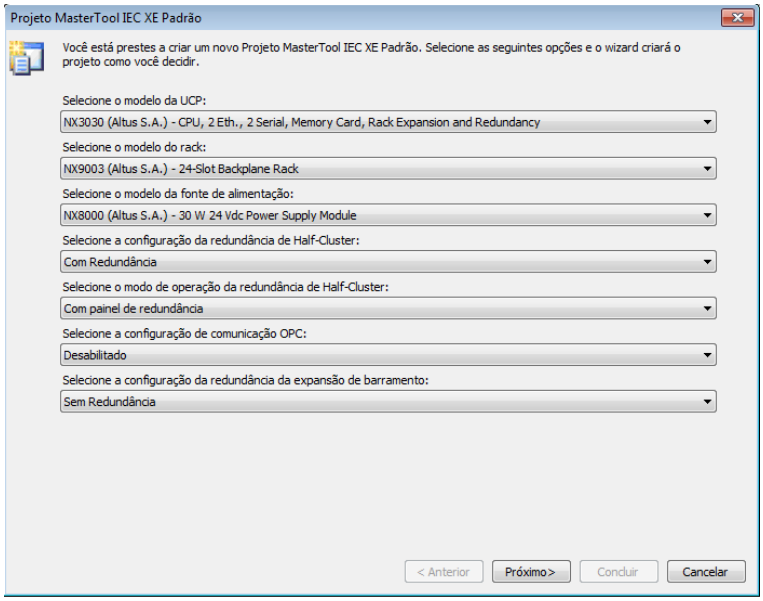
Projeto MasterTool IEC XE Padrão
Você está prestes a criar um novo Projeto MasterTool IEC XE Padrão. Selecione as seguintes opções e o wizard criará o projeto como você decidir.
Selecione o modelo da UCP:
NX3030 (Altus S.A.) - CPU, 2 Eth., 2 Serial, Memory Card, Rack Expansion and Redundancy
Selecione o modelo do rack:
NX9003 (Altus S.A.) - 24-Slot Backplane Rack
Selecione o modelo da fonte de alimentação:
NX8000 (Altus S.A.) - 30 W 24 Vdc Power Supply Module
Selecione a configuração da redundância de Half-Cluster:
Com Redundância
Selecione o modo de operação da redundância de Half-Cluster:
Com painel de redundância
Selecione a configuração de comunicação OPC:
Desabilitado
Selecione a configuração da redundância da expansão de barramento:
Sem Redundância
< Anterior
Próximo >
Concluir
Cancelar

Figura 6-16. Configuração Inicial de Hardware

Após, o usuário deve definir as redes de comunicação utilizadas na aplicação redundante:

- Selecionar o número de redes PROFIBUS: Pelo Wizard, é possível a criação de até quatro redes PROFIBUS, sendo elas simples ou redundantes. É importante salientar que esta configuração é apenas a indicada pelo Wizard, que somente propõe arquiteturas típicas. Após a conclusão do Wizard, o MasterTool permite a criação de mais redes PROFIBUS, redundantes ou únicas, desde que seja respeitado o limite de quatro módulos em cada half-cluster
- Selecionar o tipo das redes PROFIBUS:
  - Não existe (nenhum módulo NX5001 alocado)
  - Rede simples (aloca um módulo NX5001)
  - Rede redundante (aloca dois módulos NX5001)
- Selecione o tipo de rede Ethernet das interfaces da UCP
  - Rede simples com modo de falha desabilitado (não gera switchover em caso de falha)
  - Rede simples com modo de falha habilitado (gera switchover em caso de falha)
  - Rede redundante com modo de falha desabilitado (opera em conjunto com o outra interface e não gera switchover em caso de falha)
  - Rede redundante com modo de falha habilitado (opera em conjunto com o outra interface e gera switchover em caso de falha)
- Selecionar o número de redes Ethernet: Nesse caso o Wizard possibilita ao usuário criar até quatro redes simples, ou até duas redes redundantes, ou nenhuma. É importante salientar que esta configuração é apenas a indicada pelo Wizard, que somente propõe arquiteturas típicas. Após a conclusão do Wizard, o MasterTool permite a criação de mais redes Ethernet, desde que seja respeitado o limite de seis módulos em cada half-cluster.
- Selecionar o tipo de rede Ethernet:
  - Não existe (nenhum módulo NX5000 alocado)
  - Rede simples com modo de falha desabilitado (aloca um módulo NX500 e não gera switchover em caso de falha)

- Rede simples com modo de falha habilitado (aloca um módulo NX5000 e gera switchover em caso de falha)
- Rede redundante com modo de falha desabilitado (aloca dois módulos NX5000 e não gera switchover em caso de falha)
- Rede redundante com modo de falha habilitado (aloca dois módulos NX5000 e gera switchover em caso de falha)

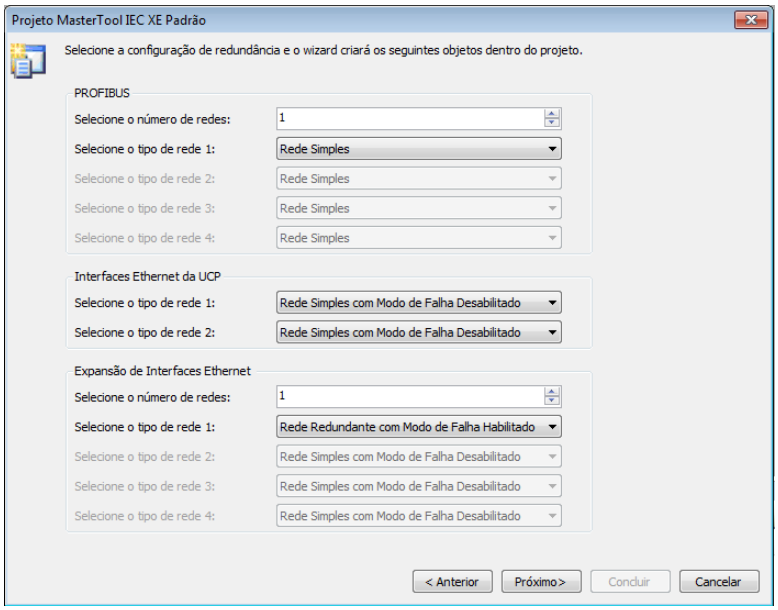
Projeto MasterTool IEC XE Padrão
Selecione a configuração de redundância e o wizard criará os seguintes objetos dentro do projeto.
PROFIBUS
Selecione o número de redes:
1
Selecione o tipo de rede 1:
Rede Simples
Selecione o tipo de rede 2:
Rede Simples
Selecione o tipo de rede 3:
Rede Simples
Selecione o tipo de rede 4:
Rede Simples
Interfaces Ethernet da UCP
Selecione o tipo de rede 1:
Rede Simples com Modo de Falha Desabilitado
Selecione o tipo de rede 2:
Rede Simples com Modo de Falha Desabilitado
Expansão de Interfaces Ethernet
Selecione o número de redes:
1
Selecione o tipo de rede 1:
Rede Redundante com Modo de Falha Habilitado
Selecione o tipo de rede 2:
Rede Simples com Modo de Falha Desabilitado
Selecione o tipo de rede 3:
Rede Simples com Modo de Falha Desabilitado
Selecione o tipo de rede 4:
Rede Simples com Modo de Falha Desabilitado
< Anterior
Próximo >
Concluir
Cancelar

Figura 6-17. Configuração das Redes de Comunicação

Então, deve ser selecionado o perfil de projeto e a linguagem padrão para a criação dos programas:

- Selecionar o perfil do projeto: Somente é possível utilizar o perfil de projeto simples para a redundância, logo a opção de seleção está desabilitada
- Selecionar a linguagem padrão para todos os programas: A linguagem selecionada pelo usuário será a padrão para todas os programas, porém pode optar por utilizar qualquer outra para uma POU específica

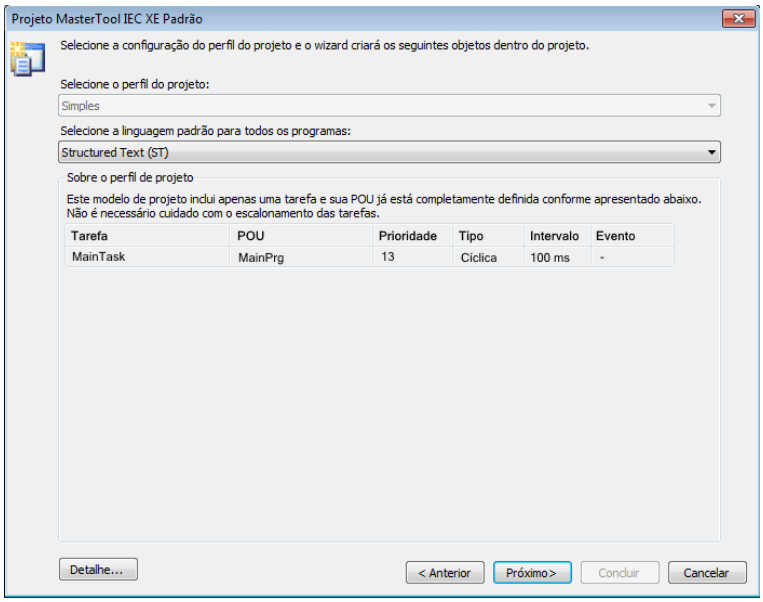

Figura 6-18. Perfil de Projeto e Linguagem Padrão

Para finalizar, o usuário deve selecionar a linguagem de programas comuns e associados à redundância:

- Programa associado à MainTask (MainPrg): Deverá ser, obrigatoriamente, em linguagem ST, sendo que o MasterTool desabilita as outras opções
- Programas associados às tarefas principais da redundância

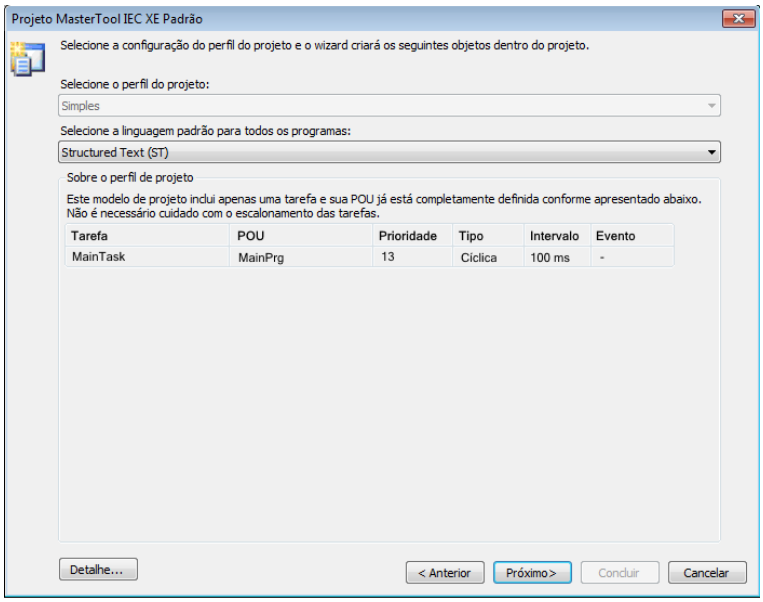

Figura 6-19. Linguagem dos Programas Específicos

ATENÇÃO: As POUs ActivePrg e NonSkippedPrg são criadas automaticamente, vazias, na linguagem selecionada nas perguntas anteriores. Em POUs criadas manualmente pelo usuário, poderá ser utilizada qualquer uma das linguagens disponíveis, exceto em POUs redundantes, que não podem ser escritas na linguagem SFC, pois esta utiliza o timer IEC em segundo plano. Para mais informações, ver a seção Limitações na Programação de um CP Redundante.

ATENÇÃO: A POU MainPrg será sempre gerada automaticamente em linguagem ST, e não pode ser alterada pelo usuário. Esta POU chama as POUs ActivePrg (somente no CP Ativo) e NonSkippedPrg (em ambos CPs).

Depois de obter respostas para as perguntas anteriores, o Wizard gera o projeto inicial, definindo um half-cluster com a seguinte configuração inicial de hardware:

- Bastidor selecionado
- Fonte de alimentação (posições 0 e 1)
- UCP NX3030 (posições 2 e 3)
- Módulo NX4010 (posições 4 e 5) e Painel PX2612, caso tenha sido selecionado.
- Após o módulo NX4010, são inseridos os módulos NX5001 necessários para implementar a rede PROFIBUS com as características definidas anteriormente pelo usuário
- Após os módulos NX5001, são inseridos os módulos NX5000 necessários para implementar a rede Ethernet com as características definidas anteriormente pelo usuário

Configuração dos Half-Clusters

O Wizard sempre é utilizado para gerar a primeira versão de um projeto redundante. Isso garante que a versão inicial do projeto será gerada rápida e corretamente.

No entanto, é possível que algumas modificações sejam necessárias em um half-cluster, tal como a inserção de novos módulos NX5001 e NX5000, que podem ser feitas alterando a tela de configuração do half-cluster. Os capítulos a seguir mostram como adicionar e/ou configurar os módulos NX5000, NX5001 e NX4010.

Algumas regras e precauções devem ser seguidas para um projeto redundante, conforme descreve a seção seguinte.

Configuração Fixa nas Posições 0 a 5 do Bastidor

Nas posições 0 a 5 do bastidor selecionado sempre devem estar instalados os seguintes módulos:

- Fonte de alimentação (posição 0)
- UCP NX3030 (posição 2)
- Módulo NX4010 (posição 4)

Estes módulos não devem ser removidos do projeto original gerado pelo Wizard.

Qualquer configuração diferente nestas posições resultará em um erro notificado pelo MasterTool na compilação do projeto.

Configurações das Portas Ethernet da UCP NX3030 (NET 1 e NET 2)

Configuração do Endereço de IP

A Figura 6-20 mostra as configurações da porta NET 1 da UCP NX3030 (a tela para configuração da porta NET 2 contém um sub-conjunto destes parâmetros). Para abrir esta tela, deve-se dar um duplo clique sobre NET 1 ou NET 2, abaixo da UCP NX3030 na árvore de dispositivos.

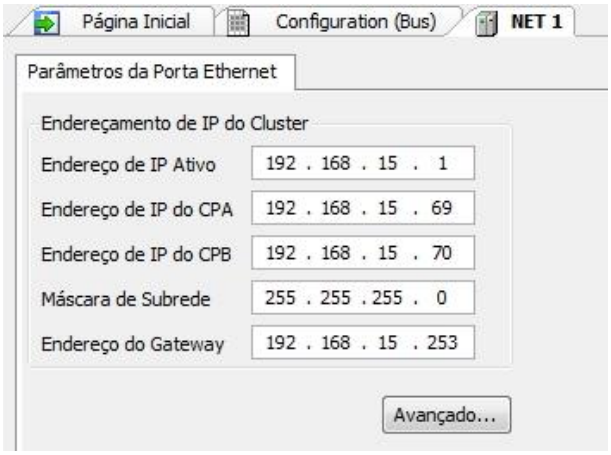
Página Inicial
Configuration (Bus)
NET 1
Parâmetros da Porta Ethernet
Endereçamento de IP do Cluster
Endereço de IP Ativo
192 . 168 . 15 . 1
Endereço de IP do CPA
192 . 168 . 15 . 69
Endereço de IP do CPB
192 . 168 . 15 . 70
Máscara de Subrede
255 . 255 . 255 . 0
Endereço do Gateway
192 . 168 . 15 . 253
Avançado...

Figura 6-20. Parâmetros da Porta Ethernet NET 1

A seguir devem ser editados os parâmetros básicos para as interfaces NET 1 e NET 2. O endereçamento será de acordo com o método de troca IP Ativo, conforme está descrito em Princípios de Funcionamento - Métodos de Troca de IP- IP Ativo.

ATENÇÃO: Os dois endereços IP das interfaces NET 1 e NET 2, bem como o Endereço do Gateway, devem pertencer à mesma sub-rede.

ATENÇÃO:
A tela de configuração de NET 2 possui a mesma estrutura da tela de configuração da NET 1, porém não contém a opção de configurações avançadas, ou seja, os parâmetros do NIC Teaming e Falha Ethernet.

NIC Teaming entre NET 1 e NET 2

A opção Avançado, na tela de configuração da interface NET 1, abre uma nova tela de configuração, a qual define se NET 1 será redundante. Caso o checkbox de Redundância de Comunicação seja marcado, as interfaces NET 1 e NET 2 formarão um par redundante com NIC Teaming, conforme descrito na seção Princípios de Funcionamento - Redes Ethernet Redundantes com NIC Teaming. Automaticamente, outros parâmetros serão habilitados e deverão ser configurados:

- Período de Teste da Redundância (ms): Período de envio do frame de teste de comunicação entre as duas NETs. Pode ser configurado com valores entre 100 e 9900
- Número de Retentativas do Teste da Redundância: Numero máximo de vezes que a NET que enviou o frame irá esperar pela resposta. Pode ser configurado com valores entre 1 e 100
- Período para Chaveamento (s): Tempo máximo que a NET Ativa irá esperar por um pacote qualquer. Pode ser configurado com valores entre 1 e 25

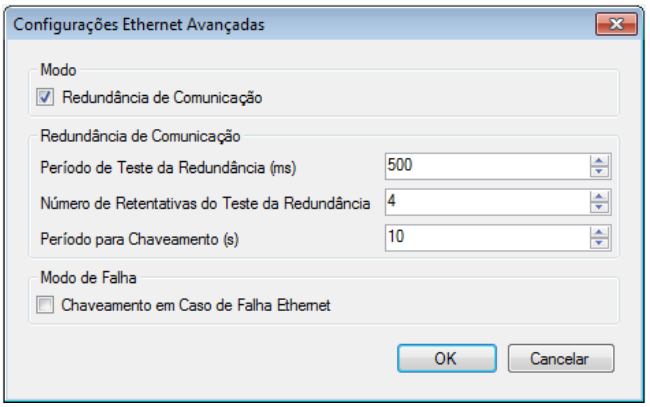

Figura 6-21. Configurações Avançadas da Ethernet

Caso o tempo de resposta do Teste da Redundância atinja Período de Teste vezes o Número de Retentativas e a interface ativa permaneça por um tempo maior que o Período para Chaveamento sem receber nenhum pacote, ocorrerá um switchover, tornando ativa a interface que até então estava inativa. É importante ressaltar que há um delay entre a detecção da falha e ativação da interface inativa devido ao tempo necessário para sua configuração. Tal delay pode chegar a algumas dezenas de milissegundos.

Quando uma das NETs estiver ativa, esta assumirá o endereço de IP configurado, permanecendo a NET inativa com seus parâmetros de Endereço de IP, Máscara de Subrede e Endereço do Gateway em branco nos diagnósticos da UCP.

ATENÇÃO:
Quando um comando de Reset Origem for executado em uma UCP configurada com NIC Teaming nas interfaces frontais NET1 e NET2, apenas a interface que estava ativa antes do comando continuará acessível. Após o comando, a interface acessível poderá ser visualizada no Menu Informativo e de Configuração da UCP.

Configuração de falha vital na NET 1 e NET 2

A opção Avançado, na tela de configuração das interfaces NET 1 e NET2, abre uma tela de configuração onde além de habilitar a redundância de comunicação também é possível configurar se a interface irá gerar um switchover em caso de falha na interface, conforme está descrito em Princípios de Funcionamento - Uso de Interfaces Ethernet com Indicação de Falha Vital.

Quando configurado em conjunto com a redundância NIC Teaming, a falha vital será considerada quando ocorrer falha nas interfaces NET1 e NET2.

Configurações dos Módulos NX5001

Inserção ou Remoção de Módulos NX5001

Pode-se inserir ou remover módulos NX5001 no bastidor do half-cluster. Para executar esta operação de forma correta, deve-se ter consciência das seguintes regras:

- O número de módulos NX5001 em cada half-cluster pode variar entre zero e quatro
- Pode-se definir até quatro redes PROFIBUS simples ou duas redes redundantes, respeitando o limite de módulos em cada half-cluster
- Quando uma rede PROFIBUS é simples, ela necessita de um único módulo NX5001 em cada half-cluster. Quando é redundante, ela necessita de 2 módulos NX5001 em cada half-cluster
- Dois módulos NX5001 utilizados para compor uma rede PROFIBUS redundante devem ocupar posições vizinhas no bastidor
- A quantidade de módulos NX5001 no bastidor deve ser compatível com o número de redes PROFIBUS existentes, e com o atributo de redundância de cada rede, ou seja:
  - 0 x NX5001: Nenhuma rede PROFIBUS
  - 1 x NX5001: Uma rede PROFIBUS simples
  - 2 x NX5001: Neste caso há duas opções:
    - Duas redes PROFIBUS simples
    - Uma rede PROFIBUS redundante
  - 3 x NX5001: Neste caso há duas opções:
    - Três redes PROFIBUS simples
    - Uma rede PROFIBUS simples e uma rede PROFIBUS redundante
  - 4 x NX5001: Neste caso há três opções:
    - Quatro redes PROFIBUS simples
    - Duas redes PROFIBUS simples e uma rede PROFIBUS redundante
    - Duas redes PROFIBUS redundantes

Depois de inserir ou remover módulos NX5001, deve-se revisar a configuração dos módulos NX5001 que restaram no bastidor.

Ajuste de Parâmetros dos Módulos NX5001

Cada módulo NX5001 utilizado em uma rede PROFIBUS simples, ou cada par redundante de NX5001 utilizado em uma rede PROFIBUS redundante, possui os seguintes parâmetros que devem ser ajustados.

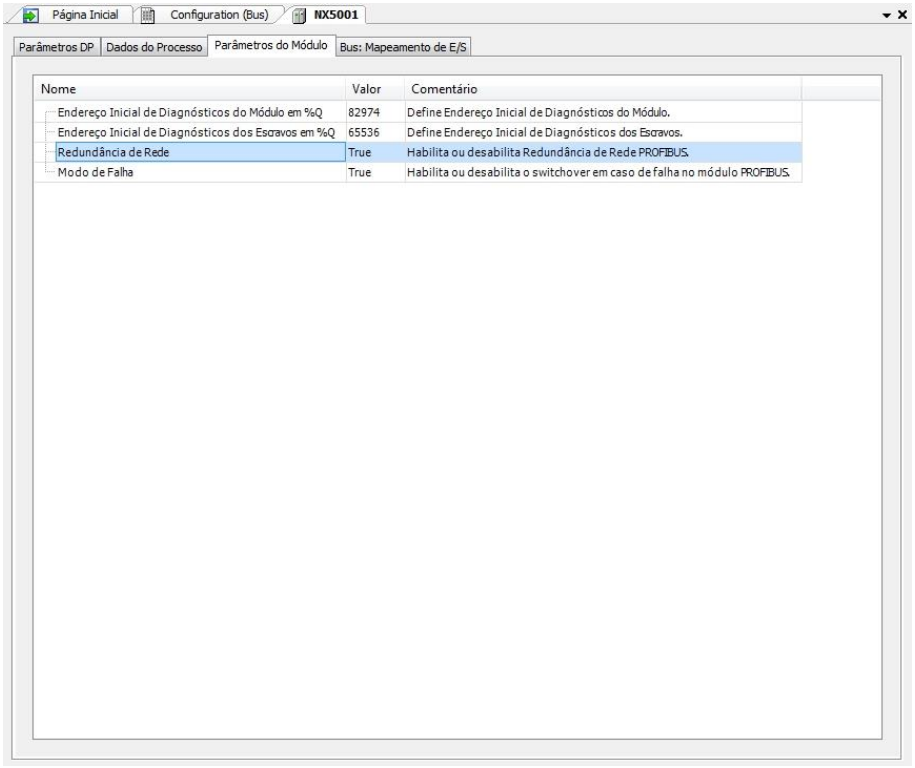

Figura 6-22. Parâmetro de Redundância NX5001

Para agrupar dois módulos NX5001 em uma rede PROFIBUS redundante, deve-se dar um duplo clique em um módulo NX5001 desagrupado que tenha outro módulo NX5001 também desagrupado à sua direita no bastidor. A seguir deve-se marcar como verdadeiro (TRUE) o parâmetro "Redundância de Rede" disponível na aba "Parâmetros do Módulo", conforme mostra Figura 6-22. Para desagrupá-los, basta seguir o mesmo procedimento, porém marcando o parâmetro como falso (FALSE). Se este parâmetro estiver marcado como verdadeiro (TRUE), os parâmetros DP e parâmetros do módulo NX5001 à sua direita estarão bloqueados para edição.

ATENÇÃO: No caso de redes redundantes, deve-se ajustar apenas os parâmetros do NX5001 presente mais a esquerda no barramento, enquanto os parâmetros do NX5001 da direita permanecem bloqueados para edição. Alguns parâmetros de uma rede são idênticos aos da outra, enquanto outros são calculados automaticamente a partir dos parâmetros da rede do NX5001 da esquerda.

É aconselhável que o endereço configurado para um mestre NX5001 em um CP redundante seja 2, pois o endereço do mestre NX5001 no CP Não-Ativo é decrementado em um, portanto o endereço do mestre NX5001 ficaria 1.

Além disso, lembra-se que:

- Os endereços de 3 a 125 são geralmente utilizados para escravos PROFIBUS
- Os endereço 0 é frequentemente utilizado para configuração e diagnóstico dos dispositivos
- O endereço 1,em especial, é reservado para ser assumido, dinamicamente, pelo mestre PROFIBUS no CP Não-Ativo (mestre PROFIBUS em modo passivo)
- O endereço 126 é muito utilizado por dispositivos escravos quando saem de fábrica
- O endereço 127 é utilizado para enviar frames em broadcast

Na próxima compilação do projeto, o MasterTool verificará possíveis erros que o usuário possa ter cometido ao inserir ou remover módulos NX5001 manualmente.

Importante salientar que, durante a execução de um projeto redundante configurado com módulos NX5001, o bit 0 de Comandos (Canal Enable Interface %QXn.0 na aba Bus: Mapeamento de E/S) é

manipulado pela aplicação redundante. As interfaces devem permanecer habilitadas durante toda a execução do programa. Desta forma, um comando executado pelo usuário para desabilitar uma interface não será executado da forma que pode se esperar. Por exemplo: se uma interface tiver o estado deste bit alterado de TRUE para FALSE em um CP Ativo, isso não será interpretado como uma falha que levaria o CP Ativo para o estado Inoperante. Neste caso, o CP Ativo permanecerá Ativo e o outro CP que irá para o estado Inoperante. Por estas razões, este bit de comando não deve ser manipulado pelo usuário em uma aplicação redundante.

Maiores detalhes sobre a configuração de redes PROFIBUS são obtidas no Manual de utilização Mestre PROFIBUS DP NX5001.

Configurações de Remotas PROFIBUS

Para configurar remotas PROFIBUS abaixo de um mestre NX5001, deve-se consultar, além do Manual de Utilização Mestre PROFIBUS DP NX5001, os seguintes manuais:

- Manual de Utilização da Série Ponto
- Manual de Utilização Cabeça PROFIBUS PO5063V1 e Cabeça Redundante PROFIBUS PO5063V5
- Manual de Utilização Cabeça PROFIBUS PO5064 e Cabeça Redundante PROFIBUS PO5065
- Manual de Utilização Rede HART sobre PROFIBUS

Para um sistema redundante deve-se atentar à configuração do parâmetro de cão-de-guarda da remota. Caso, na tela de configuração de remota, esteja habilitado o checkbox “Controle de Cão-de-Guarda”, o campo “Tempo” deverá ser configurado corretamente. Existem duas opções de configuração e o tempo utilizado deve ser o maior tempo entre:

- $WT \geq I \times 2 + 500ms$; e
- $WT \geq I \times 3$;

Onde *WT* é o tempo de cão-de-guarda e *I* é o intervalo configurado da MainTask.

Figura 6-23. Configuração de Cão-de-Guarda de uma Remota PROFIBUS

Configurações dos Módulos NX5000

Inserção ou Remoção de Módulos NX5000

Pode-se inserir ou remover módulos Ethernet NX5000 no bastidor do half-cluster. Para executar esta operação de forma correta, deve-se ter consciência de que o número de módulos NX5000 em cada half-cluster pode variar entre zero e seis. Deve-se atentar também ao fato de que módulos que formarão um par redundante NIC Teaming devem ser inseridos em posições vizinhas no bastidor.

Na próxima compilação do projeto, o MasterTool verificará possíveis erros que o usuário possa ter cometido ao inserir ou remover módulos NX5000 manualmente. Por exemplo, se o usuário inseriu mais do que 6 módulos NX5000, haverá um erro.

A interface de cada módulo será identificada com NET 1, assim como são identificadas na mecânica do produto. Caso o usuário acrescente manualmente módulos NX5000 no barramento, a identificação ocorre da mesma forma que no Wizard.

Depois de inserir ou remover módulos NX5000, deve-se revisar a configuração dos módulos NX5000 que restaram no bastidor.

Configuração dos Módulos NX5000

Para cada módulo NX5000 em um CP redundante, devem ser ajustados os parâmetros de endereço, conforme descrito na seção Princípios de Funcionamento - Métodos de Troca de IP, os quais podem ser acessados através de um duplo clique sobre a interface NET 1, abaixo de cada módulo NX5000 localizado na árvore de dispositivos.

ATENÇÃO: Caso dois módulos consecutivos formem um par redundante NIC Teaming, configura-se os parâmetros básicos somente do NX5000 da esquerda, e a edição dos parâmetros do NX5000 da direita estará bloqueada.

Agrupamento de Módulos NX5000 com Redundância NIC Teaming

Os módulos NX5000, assim como a interface NET 1 das UCPs NX3020 e NX3030, apresentam uma tela de configurações avançadas, que definirá se o módulo formará um par NIC Teaming Redundante com o módulo a sua direita. A configuração é feita conforme já descrito na seção NIC Teaming entre NET 1 e NET 2.

Para agrupar dois módulos NX5000 como um par redundante, a seguinte condição deve existir:

- Estes dois módulos NX5000 devem ocupar posições vizinhas no bastidor.

Ao fazer isto, o módulo à direita tem a edição de seus parâmetros bloqueada e os parâmetros editados no módulo à esquerda passam a ser comuns para os dois módulos.

Desmarcando o checkbox "Redundância de Comunicação" do módulo à esquerda, provoca-se a separação dos módulos, que voltam a se comportar como módulos individuais sem redundância NIC Teaming.

Configuração de Falha Vital

Os módulos NX5000, assim como as interfaces NET 1 e NET 2 permitem configurar se a interface irá gerar um switchover em caso de falha na interface, conforme está descrito em Princípios de Funcionamento - Uso de Interfaces Ethernet com Indicação de Falha Vital.

Quando configurado em conjunto com a redundância NIC Teaming a falha vital será considerada quando ocorrer falha nas dois módulos do par redundante.

Configurações do Módulo NX4010

As configurações relacionadas às variáveis %I, %Q e %M redundantes podem ser acessadas através de um duplo clique sobre o módulo NX4010, seguido da seleção da aba "Parâmetros de Redundância".

Para entender estes parâmetros, deve-se ler as seções Variáveis %I Redundantes e Não-Redundantes, Variáveis %Q Redundantes e Não-Redundantes e Variáveis %M Redundantes e Não-Redundantes.

Os seguintes parâmetros devem ser configurados:

Configuração	Descrição	Padrão de Fábrica	Possibilidades
Offset da Memória %M Redundante	**Memória (%M)**		
	Endereço inicial da memória %M redundante	0	0 (desabilitado)
Tamanho da Memória %M Redundante	Tamanho da memória %M redundante	0	0 a 65536
Offset da Memória %I Redundante	**Memória (%I)**		
	Endereço inicial da memória %I redundante	0	0 (desabilitado)
Tamanho da Memória %I	Tamanho da memória %I redundante	16384	0 a 81920

Redundante			
Offset da Memória %Q Redundante Reservado Para Drivers de E/S	**Memória (%Q)**		
	Endereço inicial da memória %Q redundante reservado para drivers de entrada e saída	0	0 (desabilitado)
Tamanho da Memória %Q Redundante Reservado para Drivers de E/S	Tamanho da memória %Q redundante reservado para drivers de entrada e saída	16384	0 a 81920
Offset da Memória %Q Redundante Reservada para Diagnósticos	Endereço inicial da memória %Q redundante reservada para a área de diagnósticos	65536	0 a 81919
Tamanho da Memória %Q Redundante Reservada para Diagnósticos	Tamanho da memória %Q redundante reservada para a área de diagnósticos	16384	0 a 81920

Tabela 6-3. Parâmetros NX4010

Configurações de I/O Drivers

A configuração de I/O drivers, em princípio, não apresenta diferenças em relação a um CP não-redundante.

O que se pode perceber é que alguns I/O drivers possuem comandos que permitem seu uso em um CP redundante, mas isto não implica em diferenças de configuração dos mesmos. Estes comandos normalmente devem ser executados no programa NonSkippedPrg. Por exemplo, um driver mestre MODBUS RTU em uma rede serial RS-485 deve ser desabilitado em um CP em estado Não-Ativo, usando código inserido pelo usuário em NonSkippedPrg. Mais informações sobre como administrar um driver MODBUS em um sistema redundante podem ser encontradas na seção Gerenciamento de Instâncias MODBUS em Sistemas Redundantes.

No caso da rede PROFIBUS, também existem comandos especiais diferenciados para os CPs em estado Ativo e Não-Ativo. Neste caso, no entanto, o gerenciamento da redundância executa tais comandos de forma automática, sem que seja necessário qualquer gerenciamento por parte do usuário.

Para configuração de E/S remoto PROFIBUS, incluindo remotas e módulos de E/S, deve-se consultar o Manual de utilização Mestre PROFIBUS DP NX5001, além da seção Configurações dos Módulos NX5001 deste manual.

Configurações da MainTask

A tela de configurações associadas à única tarefa de um CP redundante, denominada MainTask, a qual é cíclica, pode ser acessada clicando sobre MainTask na árvore de dispositivos.

Dois parâmetros devem ser ajustados nesta tela:

- Intervalo da MainTask
- Tempo de cão-de-guarda

Além disso, uma estimativa do tempo necessário para gerenciamento da redundância é calculado pelo MasterTool. Tal estimativa só é confiável depois que o projeto estiver completo, com todas POUs desenvolvidas, e áreas de memória redundantes definidas.

Diversas considerações devem ser feitas para ajustar adequadamente o intervalo da MainTask:

- O intervalo deve ser suficientemente baixo para controlar o processo efetivamente, observando os tempos de resposta de todos os laços de controle
- O intervalo deve ser suficientemente alto para acomodar, no mínimo, a soma dos dois seguintes tempos:
  - O tempo máximo de execução das POUs NonSkippedPrg e ActivePrg, em conjunto
  - O tempo necessário para gerenciar a redundância (overhead da redundância)
- Além disso, o intervalo deve ter uma folga adicional, necessária para que outros processos tenham tempo de ser executados (comunicação PROFIBUS, comunicação Ethernet com sistemas SCADA, etc...)

O MasterTool tem condições de calcular o tempo necessário para gerenciar a redundância (overhead de redundância), depois que o projeto estiver terminado (todas POUs desenvolvidas, e áreas de memória redundantes definidas).

Quanto ao tempo máximo de execução das POUs NonSkippedPrg e ActivePrg, é possível medir os tempos depois que estas POUs forem desenvolvidas. Inicialmente, o MasterTool estima 10 ms para o tempo máximo destas duas POUs em conjunto, mas o usuário deverá revisar este tempo posteriormente, em função de medidas executadas usando o projeto final.

Após cada compilação do projeto, o MasterTool soma o overhead de redundância calculado com o parâmetro que informa os tempos das POUs (NonSkippedPrg e ActivePrg), e verifica se a folga mínima parametrizada está sendo obedecida.

Por exemplo:

- Parâmetros configurados na tela da MainTask
  - intervalo da MainTask: 100 ms
  - Tempo estimado para POUs NonSkippedPrg + ActivePrg: 10 ms
  - Folga Mínima: 30%
- Overhead calculado para redundância: 50 ms

Neste caso, o tempo total utilizado é de 60 ms (10 ms + 50 ms), o que consiste em 60% do ciclo da MainTask (100 ms). Desta forma, a folga é de 40%, e portanto a folga mínima de 30% está sendo respeitada.

ATENÇÃO: Um erro de compilação pode ser gerado caso a folga mínima não seja respeitada. Desde que esteja configurado nos Parâmetros do Projeto da UCP.

ATENÇÃO: Havendo sucesso ou não na compilação, o MasterTool informa se a folga é respeitada e o cálculo do overhead de redundância na janela de mensagens.

Programa ActivePrg

Nesta POU o usuário deve criar a aplicação principal, responsável pelo controle de seu processo. Esta POU é chamada pela POU principal (MainPrg), sendo executada apenas no CP Ativo.

O usuário pode também criar POUs adicionais (programa, funções ou bloco funcional), e chamá-las ou instanciá-las dentro da POU ActivePrg, para fins de estruturação de seu programa. Também é possível chamar funções e instanciar blocos funcionais definidos em bibliotecas.

Deve-se lembrar que todas as variáveis simbólicas definidas na POU ActivePrg, assim como instâncias de blocos funcionais definidas, serão variáveis redundantes.

Variáveis simbólicas definidas em POUs adicionais do tipo programa que forem chamadas dentro de ActivePrg também serão variáveis redundantes.

ATENÇÃO:
Variáveis do tipo VAR_TEMP não devem ser utilizadas no programa redundante.

Programa NonSkippedPrg

Esta POU é destinada para controles que devem ser executados em ambos CPs (CPA e CPB), independente do seu estado de redundância. Esta POU também é chamada pela POU principal (MainPrg).

Deve-se lembrar que todas as variáveis simbólicas definidas na POU NonSkippedPrg, assim como instâncias de blocos funcionais definidas, serão variáveis não-redundantes.

O usuário pode também criar POUs adicionais (programa, funções ou bloco funcional), e chamá-las ou instanciá-las dentro da POU NonSkippedPrg, para fins de estruturação de seu programa. Também é possível chamar funções e instanciar blocos funcionais definidos em bibliotecas.

ATENÇÃO:
Quando chamar POUs adicionais do tipo programa dentro de NonSkippedPrg, desmarque as mesmas na janela Redundancy Configuration do MasterTool. Por padrão as variáveis simbólicas declaradas dentro destas POUs serão redundantes e, dentro da NonSkippedPrg, normalmente deseja-se variáveis não-redundantes. Normalmente o código de NonSkippedPrg é pequeno, dispensando a chamada de POUs adicionais do tipo programa para sua estruturação, mas, caso a sua estruturação seja necessária o mais recomendável é utilizar bloco funcional ou funções.

Exemplos típicos de controles executados na NonSkippedPrg são os seguintes:

- Criar uma estrutura compacta de diagnósticos (%Q) a serem reportados para um sistema SCADA, a partir de uma estrutura completa de diagnósticos, onde diversos diagnósticos não são de interesse para o sistema SCADA. Estes diagnósticos podem ser extraídos de estruturas de dados como RedDgnLoc, RedDgnRem, RedUsrLoc, RedUsrRem, etc
- Copiar comandos recebidos de um SCADA para os respectivos campos da estrutura de dados RedCmdLoc, e intertravar estes comandos se necessário
- Gerenciar switchovers controlados pelo usuário, em função de falhas não-vitais tais como a comunicação com um sistema SCADA ou com dispositivos MODBUS
- Habilitar ou desabilitar alguns I/O drivers específicos, dependendo do estado da redundância (Ativo ou Não-Ativo). Por exemplo, um driver mestre MODBUS RTU em um barramento RS-485 deve ser desabilitado no CP Não-Ativo. Para mais informações, consultar a seção Gerenciamento de Instâncias MODBUS em Sistemas Redundantes

ATENÇÃO:
Não recomenda-se utilizar os blocos funcionais TOF_RET, TON_RET, TOF e TON no programa NonSkippedPrg. Ver seção Limitações na Programação de um CP Redundante.

Objeto Redundancy Configuration

Este objeto, localizado na árvore de dispositivos do projeto, é criado automaticamente pelo Wizard.

Ele é utilizado para determinar quais POUs e GVLs são redundantes, e portanto sincronizadas entre os CPs. Por padrão, POUs e GVLs criadas pelo usuário são marcadas como redundantes, cabendo a ele reverter a marcação quando necessário.

ATENÇÃO:
Os objetos PV, PIDControl e PidRetainGVL não podem ser marcados individualmente. Caso seja necessário que estes sejam alterados, deve ser marcada a opção Selecionar Todos.

GVL Diagnostics

Esta GVL especial é criada e preenchida automaticamente pelo Wizard, e não pode ser modificada pelo usuário.

Diagnósticos e comandos do sistema, inclusive das estruturas de dados de redundância (RedDgnLoc, RedDgnRem, RedCmdLoc, RedCmdRem), residem em variáveis de representação direta %Q ou %I especiais.

A GVL Diagnostics contém diversas sentenças com a diretiva AT para definir nomes simbólicos para estes diagnósticos e comandos. Desta maneira, quando o usuário precisar referenciar estas variáveis, poderá usar um nome simbólico ao invés de uma referência numérica.

GVLs com Variáveis Simbólicas Redundantes

O usuário pode criar outras GVLs, diferentes das citadas anteriormente, para declarar variáveis simbólicas redundantes. Para isso, a GVL deve estar marcada na configuração do objeto Redundancy Configuration, na árvore de dispositivos do projeto, significando que essa GVL é redundante. Por padrão, todas as GVLs criadas pelo usuário serão, inicialmente, redundantes.

ATENÇÃO: Por boa prática, é recomendado evitar a utilização da diretiva AT em GVLs que contenham declarações de variáveis simbólicas redundantes, para evitar o mapeamento de variáveis em áreas não-redundantes.

POUs do Tipo Program com Variáveis Simbólicas Redundantes

O usuário pode declarar variáveis simbólicas redundantes em POUs do tipo programa, com exceção da POU NonSkippedPrg onde as variáveis simbólicas declaradas são consideradas não-redundantes.

Para definir uma nova POU como redundante deve-se, após a criação da POU, marcá-la na configuração do objeto Redundancy Configuration, na árvore de dispositivos do projeto. Por padrão, todas as POUs criadas pelo usuário serão, inicialmente, redundantes.

ATENÇÃO: Por boa prática, é recomendado evitar a utilização da diretiva AT em POUs que sejam redundantes, afim de evitar o mapeamento de variáveis em áreas não-redundantes.

Utilização de Breakpoints em Sistemas Redundantes

Para sistemas redundantes, o recomendado é utilizar breakpoints apenas no half-cluster Ativo, com o outro half-cluster desligado. Caso contrário, quando a execução da aplicação atingir um breakpoint, o half-cluster Reserva irá assumir o estado Ativo, desligando o CP ativo.

Gerenciamento de Instâncias MODBUS em Sistemas Redundantes

Para o caso de utilização da falha vital das interfaces Ethernet estar desabilitada ou no caso de instâncias MODBUS Server, as instâncias MODBUS são independentes da redundância e, portanto, deverão ser gerenciadas na aplicação, ficando a critério do usuário quais instâncias deverão ser habilitadas/desabilitadas quando um CP entrar em um estado Não-Ativo. Quando for habilitada a falha vital para portas Ethernet com MODBUS Client não é necessário implementar código adicional para controlar o switchover.

O exemplo abaixo, inserido no programa NonSkippedPrg, faz a verificação do estado atual do CP e, caso esteja em um estado Não-Ativo, desabilita as instâncias MODBUS RTU Mestre e Escravo e a instância MODBUS Ethernet Servidor:

```
VAR
   eRedStateLocal : REDUNDANCY_STATE;
   eRedStateLocal_old : REDUNDANCY_STATE;
END_VAR

// Leitura do estado atual do CP local
eRedStateLocal := DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.eRedState;
```

```
// O estado do CP local mudou?
IF eRedStateLocal <> eRedStateLocal_old THEN

    IF eRedStateLocal = REDUNDANCY_STATE.ACTIVE THEN
        // CP local entrou em estado Ativo
        Diagnostics.DG_MODBUS_RTU_Slave.tCommand.bRestart := TRUE;
        Diagnostics.DG_MODBUS_RTU_Master.tCommand.bRestart := TRUE;
        Diagnostics.DG_MODBUS_Server.tCommand.bRestart := TRUE;
    ELSE
        // CP local entrou em um estado Não Ativo
        Diagnostics.DG_MODBUS_RTU_Slave.tCommand.bStop := TRUE;
        Diagnostics.DG_MODBUS_RTU_Master.tCommand.bStop := TRUE;
        Diagnostics.DG_MODBUS_Server.tCommand.bStop := TRUE;
    END_IF
        // Salva o último estado do CP local
        eRedStateLocal_old:= eRedStateLocal;
END_IF
```

Limitações na Programação de um CP Redundante

Em um CP redundante, irão existir algumas limitações quanto à programação dos half-clusters. Estas limitações são tratadas nas subseções abaixo.

Limitações em GVLs e POUs Redundantes

Em uma GVL ou em uma POU do tipo programa que sejam redundantes, as seguintes limitações devem ser respeitadas para um correto funcionamento dos half-clusters:

- Não utilizar variáveis do tipo VAR_TEMP
- Não misturar tipos de variáveis (VAR, VAR RETAIN, VAR PERSISTENT, etc...), devendo ser utilizado somente um dos tipos em cada GVL ou POU
- Não misturar declaração de variáveis simbólicas com ATs nas GVLs. Devem ser criadas GVLs separadas, declarando em uma as variáveis do tipo AT e em outra as variáveis simbólicas
- Não armazenar o endereço de uma variável em uma variável redundante (fazer de uma variável redundante um ponteiro para um endereço), pois os endereços das variáveis podem ser diferentes no CPA e no CPB
- Não utilizar blocos funcionais de escrita e leitura ao relógio RTC em POUs redundantes. Mais detalhes podem ser encontrados no capítulo Relógio RTC

Limitações no Programa Não-Redundante (NonSkippedPrg)

Em uma POU do tipo programa que não seja redundante, no caso, a POU NonSkippedPrg, as seguintes limitações devem ser respeitadas para um correto funcionamento dos half-clusters:

- Não podem ser utilizados os blocos funcionais TON e TOF tradicionais, pois os mesmos utilizam o timer IEC. Quando o CP Reserva entrar em estado ativo (com o outro half-cluster saindo do estado Ativo), o timer IEC será sincronizado, causando uma descontinuidade no valor do timer. Deve-se optar pelos blocos funcionais TON_NR e TOF_NR disponibilizados na library NextoStandard. Ver capítulo Configuração - Blocos Funcionais - Timer Não-Redundante
- Não podem ser utilizados POUs do tipo programa escritas na linguagem SFC (Sequenciamento Gráfico de Funções), pois estas utilizam o timer IEC para temporizar as transições
- Não misturar declaração de variáveis simbólicas com ATs nas GVLs. Devem ser criadas GVLs separadas, declarando em uma as variáveis do tipo AT e em outra as variáveis simbólicas

Obtendo o Estado da Redundância de um Half-Cluster

É possível verificar o estado da redundância de um half-cluster na Estrutura de Diagnósticos da Redundância:

```
VAR
eRedStateLocal : REDUNDANCY_STATE;
```

```
END_VAR

eRedStateLocal := DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.eRedState;
```

Deste modo, o usuário poderá fazer o controle de lógicas que dependam do estado redundante do CP.

Leitura de Diagnósticos Não-Redundantes

Um projeto redundante, além de apresentar diagnósticos redundantes (Estrutura de Diagnósticos da Redundância ou os diagnósticos de uma remota PROFIBUS), apresenta também diagnósticos que não são redundantes (diagnósticos específicos dos módulos NX5000, NX5001, NX3030, etc.). Estes diagnósticos não-redundantes poderão não ser válidos nos primeiros instantes após o CP assumir o estado Ativo, pois os mesmos não são sincronizados com o outro CP (não se sabe o estado em que estava o diagnóstico quando o CP remoto estava Ativo). Portanto, estes diagnósticos devem ser desconsiderados nos primeiros instantes em estado Ativo, até que tenham valores válidos. Tipicamente o tempo durante o qual os diagnósticos devem ser desconsiderados é de 5 s.

O exemplo abaixo mostra como desconsiderar os diagnósticos bSlaveNotPresent e bPbusCommFail do módulo Mestre PROFIBUS NX5001 durante os primeiros 5 s em que o CP está em estado Ativo.

Lógica em NonSkippedPrg:

```
PROGRAM NonSkippedPrg
VAR
   TON_DiagEnable : TON_NR;
   bDiagEnable : BOOL;
   bIsActiveState : BOOL;
   bIsActiveState_old : BOOL;
END_VAR

bIsActiveState := (DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.eRedState
= REDUNDANCY_STATE.ACTIVE);
TON_DiagEnable(IN:= (bIsActiveState = bIsActiveState_old), PT:= T#5S, Q=>
bDiagEnable);
bIsActiveState_old := bIsActiveState;
```

Lógica em ActivePrg:

```
IF NonSkippedPrg.bDiagEnable THEN
   IF DG_NX5001.tGeneral.bSlaveNotPresent OR
   DG_NX5001.tGeneral.bPbusCommFail THEN
      //Executa ações de diagnóstico ativo
   END_IF
END_IF
```

Carga de Programas em um CP Redundante

A seção Programação de um CP Redundante tratou de aspectos relativos ao desenvolvimento de um projeto para um CP redundante com UCP NX3030.

Nesta seção, discutem-se métodos e etapas para carregar este projeto em um CP redundante, considerando diversas situações, tais como:

- Carga do projeto em uma UCP NX3030 nova retirada da caixa, ou em uma UCP contendo um projeto desconhecido
- Carga on-line de modificações
- Carga off-line de modificações com interrupção do controle do processo, durante uma parada programada do processo
- Carga off-line de modificações sem interrupção do controle do processo, valendo-se de características da redundância

Carga Inicial de um Projeto Redundante

Esta seção descreve os passos necessários para fazer a primeira carga de projeto redundante em uma UCP NX3030. Isto é necessário, por exemplo, para uma UCP nova de fábrica, ou para uma UCP que contenha um projeto desconhecido.

ATENÇÃO: Os passos seguintes devem ser executados para os dois half-clusters (CPA e CPB) que compõem um CP redundante. Primeiro deve-se executar todos os passos para um dos CPs, e depois para o outro.

Primeiro Passo - Descoberta do Endereço IP para Conexão do MasterTool

O primeiro é descobrir o endereço IP do canal NET 1 desta UCP, para conexão ao MasterTool.

Isto deve ser feito através do visor e botão da UCP NX3030, conforme descrito em Menu Informativo e de Configuração da UCP. O menu REDE informa os endereços IPs que podem ser utilizados para conexão via MasterTool.

Segundo Passo – Verificar Conflito de Endereços IP

Antes de executar o próximo passo, deve-se ter certeza de que não existe, na rede, outro equipamento com o mesmo endereço IP descoberto no primeiro passo. Isso pode ser descoberto, por exemplo, desconectando o CP da rede e executando um comando "ping" no seu endereço IP. Como o CP está desconectado, espera-se que este "ping" falhe. Se o "ping" funcionar, existe outro equipamento com o mesmo endereço IP.

Caso o endereço IP já esteja em uso por outro equipamento na rede, deve-se executar o terceiro passo, e também alguns passos seguintes, utilizando um cabo de crossover para conectar diretamente o PC com o software MasterTool IEC XE ao CP, evitando assim conflitos de endereços IP. Em um dos passos seguintes, ao carregar o projeto no CP, serão atualizados os endereços IP definitivos do CP (ver seção Configurações das Portas Ethernet da UCP NX3030 (NET 1 e NET 2)).

Terceiro Passo – Preparar Conexão do MasterTool (Definir Caminho Ativo)

O terceiro passo consiste em dar um duplo clique sobre o Device (NX3030) na árvore de dispositivos, entrar na aba "Configurações de Comunicação", clicar sobre o Gateway, e pressionar o botão "Mapear Rede" para listar todos os CPs detectados pelo MasterTool na rede.

Neste momento, espera-se encontrar uma lista de CPs cuja identificação contém o endereço IP descoberto no primeiro passo e o tipo da UCP (NX3030). Caso o usuário tenha trocado anteriormente o nome da UCP na rede, este nome será o visualizado neste momento. A seção Conexão do MasterTool com uma UCP NX3030 de um CP Redundante descreve com maiores detalhes as possíveis identificações que podem ser observadas nesta lista. De qualquer maneira, todas as possíveis identificações possuem um campo que mostra o endereço IP ou parte dele. Por exemplo, os bytes que estão entre colchetes formam o endereço da UCP. O byte que está a direita, entre os bytes que estão entre colchetes, indica o final do endereço IP em hexadecimal. Se os bytes formarem o seguinte endereço [0010], isto significa que o byte com o valor 10 indica que o final do endereço IP dessa UCP é xxx.xxx.xxx.16. A seguir, deve-se clicar sobre este CP na lista e pressionar o botão "Definir Caminho Ativo". Feito isso, o CP selecionado deve aparecer em negrito na lista, o que indica que o MasterTool está preparado para conectar-se a este CP.

Quarto Passo – Identificar a UCP NX3030 e Conferir no Visor da UCP

O quarto passo consiste em identificar o half-cluster, como CPA ou CPB. Isto é feito através do menu Comunicação / Configuração Básica de Cluster:

A seguir, a combo-box "Identificação do CP" permite selecionar uma das três seguintes opções:

- CPA
- CPB
- Não-Redundante

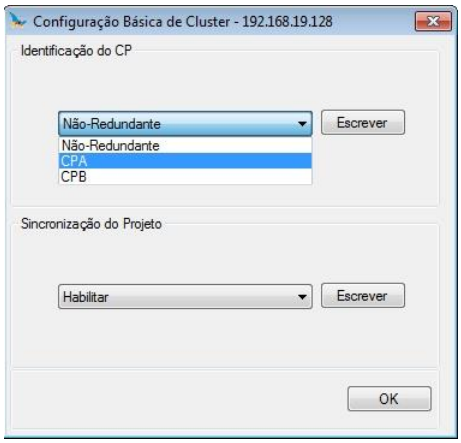

Figura 6-24. Identificação do CP

No caso de um CP redundante, o usuário deve selecionar CPA ou CPB. Depois de selecionar a opção desejada, o botão "Escrever" ao lado desta combo-box deve ser pressionado. O MasterTool retornará um aviso de que a UCP será reinicializada e se o usuário deseja confirmar a ação. Após, irá aparecer uma mensagem indicando sucesso ou falha do comando e, em caso de sucesso, a UCP será reinicializada.

ATENÇÃO:
A UCP NX3030 não pode estar em modo Run quando este comando é executado. Antes de executar o comando, o usuário deve colocar a UCP em Modo Stop. Caso a UCP esteja em modo Run, o comando não é executado e o MasterTool avisa que o comando falhou.

Logo depois de executar o comando de identificação com sucesso, observa-se que a identificação selecionada aparece nos Diagnósticos da Redundância no Visor Gráfico da UCP NX3030.

A identificação do CP também estará disponível em um diagnóstico interno (DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.ePLC_ID). Este diagnóstico é atualizado a partir da memória não-volátil a cada ciclo da MainTask, portanto é necessário que o CP volte ao modo Run para que o mesmo seja atualizado. Abaixo, estão relacionados os códigos retornados pelo diagnóstico e suas respectivas descrições:

- Não-Redundante: 0
- CPA: 2
- CPB: 3

A identificação de um CP não faz parte do projeto redundante desenvolvido com o MasterTool. Tal identificação reside apenas em uma área de memória não-volátil da UCP, que pode ser modificada usando o MasterTool.

CUIDADO:
A redundância não funcionará adequadamente caso uma das UCPs não seja identificada como CPA e a outra UCP não seja identificada como CPB, podendo ocasionar interrupção do controle do processo. Caso uma UCP NX3030 deva ser substituída (ex: após um dano), a nova UCP deve ser previamente identificada com a mesma identificação daquela UCP que está sendo substituída. Deve-se usar o visor da UCP para verificar se os dois CPs foram corretamente identificados.

Quinto Passo – Carga do Projeto Redundante

Este passo descreve a carga do projeto redundante no CP. Este projeto deve ser preparado conforme descrito na seção Programação de um CP Redundante.

Um projeto simples (básico) pode ser preparado seguindo, no mínimo, as seguintes subseções desta seção:

- Wizard para Criação de um Novo Projeto Redundante
- Configurações das Portas Ethernet da UCP NX3030 (NET 1 e NET 2)

Obviamente, também é possível fazer um projeto completo e só depois carregá-lo no CPA e CPB, por exemplo, caso o hardware destes CPs não esteja disponível durante o desenvolvimento do projeto com o MasterTool.

A primeira carga de um projeto redundante em uma UCP, previamente identificada como CPA ou CPB, ainda deve ser feita utilizando aquele endereço IP descoberto no primeiro passo deste procedimento, e selecionado no terceiro passo deste procedimento.

A carga do projeto é feita através do menu Comunicação / Login.

ATENÇÃO:
Dentro do projeto desenvolvido com o MasterTool IEC XE e carregado no CP neste passo, foram definidos novos endereços IP para a interface NET 1 do CPA e CPB (Endereço de IP do CPA e Endereço de IP do CPB), assim como um endereço IP para a interface NET 1 do CP Ativo (Endereço de IP Ativo) – ver seção Configurações das Portas Ethernet da UCP NX3030 (NET 1 e NET 2).

Portanto, depois desta carga inicial, aquele endereço IP descoberto no primeiro passo deste procedimento normalmente não será mais válido. Esta mudança do endereço IP em NET 1 ou NET2 provocará uma perda da conexão do MasterTool com o CP, que será notificada.

Para maiores detalhes sobre como reconectar o MasterTool, ver seção Conexão do MasterTool com uma UCP NX3030 de um CP Redundante.

Conexão do MasterTool com uma UCP NX3030 de um CP Redundante

Depois de executar o procedimento descrito na seção Carga Inicial de um Projeto Redundante nos dois CPs (CPA e CPB), a conexão ao MasterTool, através da interface NET 1 da UCP NX3030, poderá ser feita através de um dos seguintes endereços:

- Endereço de IP do CPA: endereço de NET 1 exclusivo para o CPA
- Endereço de IP do CPB: endereço de NET 1 exclusivo para o CPB

Independente do estado do CP, o MasterTool só consegue se conectar ao mesmo utilizando o endereço exclusivo do CP, configurado em Endereço de IP do CPX. Porém, caso o CP se encontre em estado Ativo, todos os outros serviços poderão se conectar ao CP tanto pelo Endereço de IP do CPX quanto pelo Endereço de IP Ativo.

Para conectar-se a determinado CP, em primeiro lugar deve-se dar um duplo clique sobre o Device (NX3030) na árvore de dispositivos, entrar na aba “Configurações de Comunicação”, clicar sobre o Gateway, e pressionar o botão “Mapear Rede” para listar todos os CPs detectados pelo MasterTool na rede.

Nesta lista, será possível encontrar as seguintes identificações padrão, caso o nome do CP na rede não tenha sido alterado anteriormente pelo usuário:

- NX3030_<IP address>_PLCA: identificação relacionada ao CPA. Neste caso, o campo <IP address> deve coincidir com o endereço Endereço de IP do CPA configurado no projeto.
- NX3030_<IP address>_PLCB: identificação relacionada ao CPB. Neste caso, o campo <IP address> deve coincidir com o endereço Endereço de IP do CPB configurado no projeto.

A seguir, deve-se selecionar o CP desta lista onde o MasterTool deve se conectar, e pressionar o botão “Definir Caminho Ativo”. Posteriormente, ao executar o comando do menu Comunicação / Login, o MasterTool se conectará a este CP.

ATENÇÃO:
O MasterTool só consegue se conectar a um CP de cada vez. Para conectar-se a diversos CPs, deve-se abrir múltiplas instâncias do MasterTool, cuidando sempre para abrir o projeto correto em cada instância.

Carga de Modificações em um Projeto Redundante

Depois que os dois CPs (CPA e CPB) que formam um CP redundante já receberam uma carga inicial, conforme descrito na seção Carga Inicial de um Projeto Redundante, é possível carregar modificações sucessivas do projeto, à medida que tais modificações forem necessárias.

A conexão do MasterTool aos CPs para executar a carga de modificações deve ser feita conforme descreve-se na seção Conexão do MasterTool com uma UCP NX3030 de um CP Redundante. Nesta seção explica-se como é possível conectar-se a um CP específico (CPA ou CPB), ou ao CP Ativo, ou ao CP Não-Ativo.

Normalmente as modificações devem ser carregadas no CP Ativo, e a seguir serão automaticamente sincronizadas para o CP Não-Ativo, através dos canais de sincronismo NETA/NETB. Portanto, o MasterTool normalmente deve usar o endereço de IP exclusivo do CP Ativo (Endereço de IP do CPX), para se conectar ao canal NET 1 da UCP NX3030 do CP Ativo. Para verificar qual dos CPs está em estado Ativo, poderá ser seguido o mesmo passo descrito em Carga Inicial de um Projeto Redundante - Quarto Passo – Identificar a UCP NX3030 e Conferir no Visor da UCP.

ATENÇÃO:
Carregar um projeto no CP Não-Ativo normalmente é inútil, pois a sincronização automática de projetos (do CP Ativo para o CP Não-Ativo) cancelaria o efeito desta carga. Entretanto, há situações especiais em que a sincronização de projetos pode ser desabilitada temporariamente, sendo possível e útil carregar um projeto diferente no CP Não-Ativo. Estas situações especiais são discutidas na seção Explorando a Redundância para Carga Off-Line de Modificações sem Interrupção do Controle do Processo.

Carga de Modificações Off-Line e On-Line

Modificações de projeto podem ser carregadas off-line ou on-line.

Cargas off-line requerem a parada do CP onde a modificação deve ser carregada. Por outro lado, cargas on-line permitem que o CP continue executando sua aplicação enquanto a modificação é carregada.

Alguns tipos de modificações exigem carga off-line, ou seja, não podem ser carregadas on-line no CP onde o MasterTool está conectado. Neste caso, existem duas opções:

- Interromper o controle do processo, executando o procedimento descrito na seção Carga Off-Line de Modificações com Interrupção do Controle do Processo
- Valer-se da redundância do CP e das redes PROFIBUS para não interromper o controle do processo, mesmo com a necessidade de fazer cargas off-line em cada um dos half-clusters (CPA ou CPB). Um procedimento para atingir este objetivo é descrito na seção Explorando a Redundância para Carga Off-Line de Modificações sem Interrupção do Controle do Processo

Modificações que Demandam Carga Off-Line com Interrupção do Controle do Processo

As seguintes modificações em um projeto farão com o que o mesmo se torne impossível de ser carregado num sistema redundante sem a interrupção do controle do processo:

- Modificações nas áreas de memória redundantes (alteração dos Parâmetros de Redundância do módulo NX4010)

ATENÇÃO:
Não será possível alterar o tamanho das áreas de memória redundantes sem interromper o controle do processo. Portanto estas áreas devem ser cuidadosamente planejadas e configuradas previamente.

Modificações que Demandam Carga Off-Line

As seguintes modificações demandam carga off-line no CP:

- Adicionar ou remover dispositivos na árvore de dispositivos, por exemplo:
  - Módulos no bastidor principal (mestres PROFIBUS NX5001, interfaces Ethernet NX5000, etc.)
  - Remotas em redes PROFIBUS
  - Módulos de E/S em remotas em redes PROFIBUS
  - Instâncias MODBUS
- Modificar parâmetros dentro de dispositivos existentes na árvore de dispositivos, por exemplo:
  - Endereços IP e outros parâmetros de interfaces Ethernet
  - Parâmetros de mestres PROFIBUS
  - Parâmetros de remotas PROFIBUS
  - Parâmetros de módulos de E/S dentro de remotas PROFIBUS
- Modificações nas configurações das tarefas.
- Atualização de Projeto devido a Atualização do programador MasterTool IEC XE

Modificações que Permitem Carga On-Line

Em princípio, as modificações não citadas nas seções Modificações que Demandam Carga Off-Line com Interrupção do Controle do Processo e Modificações que Demandam Carga Off-Line, permitem carga on-line.

Mesmo assim, a seguir serão citadas as principais modificações que permitem carga on-line no CP onde o MasterTool está conectado. As modificações citadas abaixo valem para variáveis, POUs e GVLs, redundantes ou não.

- Adicionar POUs do tipo programa, desde que estas POUs não precisem ser associadas a alguma tarefa
- Remover POUs do tipo programa, desde que estas POUs não estejam associadas a alguma tarefa.
- Adicionar ou remover POUs do tipo função ou bloco funcional
- Modificar o código de quaisquer tipos de POU (programa, função ou bloco funcional)
- Adicionar ou remover variáveis simbólicas em quaisquer tipos de POU (programa, função ou bloco funcional, sendo elas redundantes ou não)
- Adicionar ou remover instâncias de bloco funcional em POUs do tipo programa ou bloco funcional
- Adicionar ou remover GVLs
- Adicionar ou remover variáveis simbólicas ou instâncias de bloco funcional em GVLs

Carga On-Line de Modificações

Na seção Carga de Modificações Off-Line e On-Line, foram descritas modificações que demandam carga off-line e aquelas que permitem carga on-line.

Uma carga on-line deve ser feita conectando o MasterTool ao canal NET 1 do CP Ativo, utilizando seu endereço exclusivo de IP. Antes da versão 2.01 do MasterTool IEC XE, era necessário que o usuário selecionasse a opção “Criar Aplicação de Inicialização” no menu Comunicação, após o envio da aplicação, para que as modificações fossem enviadas para a área de memória não-volátil da UCP e pudessem ser sincronizadas. A partir da versão 2.01 esta operação não é mais necessária. Após o envio da aplicação a operação de envio para a memória não volátil é realizada de forma automática.

ATENÇÃO: É importante lembrar que modificações online, sem que a opção mencionada anteriormente seja selecionada, serão perdidas no caso de um Reset a Quente ou a UCP ser desligada.

ATENÇÃO:
Uma alteração online na declaração de variáveis retentivas da aplicação (inclusão ou remoção de variáveis) seguida de uma queda da alimentação do CP antes de "Criar Aplicação de Inicialização", irá corromper a memória retentiva, pois o valor das variáveis retentivas que foram salvas e recuperadas não correspondem as variáveis da aplicação em memória e que foi restaurada.

Quando o CP Não-Ativo percebe que possui um projeto diferente do CP Ativo, ele executa as seguintes ações:

- Negocia uma sincronização automática de projeto com o CP Ativo
- Caso esteja no estado Reserva ou Inicializando, chaveia para o estado Não-Configurado, e permanece neste estado até que os projetos estejam novamente sincronizados. Depois disso, volta automaticamente ao estado Reserva
- Caso esteja no estado Não-Configurado ou Inativo, o botão STAND-BY do painel PX2612 deve ser pressionado, ou comando equivalente a este botão deve ser executado. Desta forma, depois da sincronização do projeto, sairá do estado Não-Configurado e poderá ir ao estado Reserva, ou voltar para o estado Inativo se ocorrer alguma falha

Carga Off-Line de Modificações com Interrupção do Controle do Processo

Nesta seção, define-se o procedimento para executar uma carga off-line que interromperá o controle do processo. Tal situação é aceitável em determinados tipos de processos e durante paradas programadas deste processo.

Uma carga off-line deste tipo deve ser feita conectando o MasterTool ao canal NET 1 do CP Ativo, utilizando o endereço exclusivo do CP em estado Ativo (Endereço de IP do CPX). Antes de iniciar uma carga off-line no CP Ativo, o usuário recebe o seguinte aviso do MasterTool:

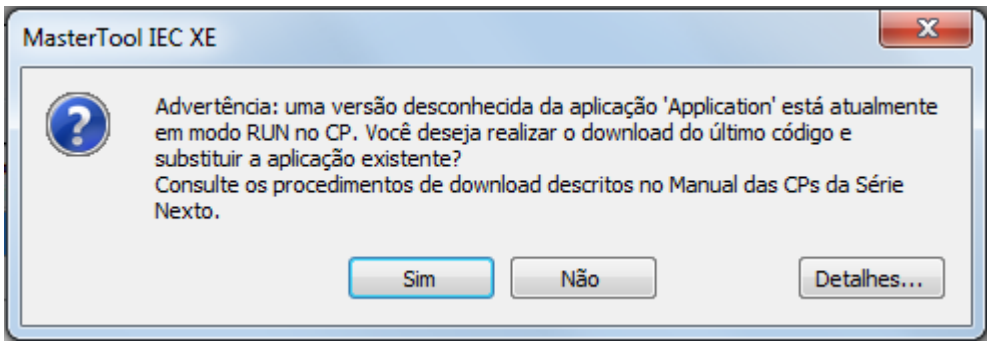
MasterTool IEC XE
Advertência: uma versão desconhecida da aplicação 'Application' está atualmente em modo RUN no CP. Você deseja realizar o download do último código e substituir a aplicação existente?
Consulte os procedimentos de download descritos no Manual das CPs da Série Nexto.
Sim
Não
Detalhes...

Figura 6-25. Aviso de Carga Off-Line

Ao pressionar o botão *Sim* o projeto é enviado. Quando uma carga off-line é executada, o controle do processo é interrompido, pois o envio do projeto deve ser feito para o CP Ativo que sairá do modo de execução (Run), e portanto estará no estado Não-Configurado.

Outro ponto importante é que se o outro CP estiver em estado Reserva, deve-se passá-lo para o estado Inativo, por exemplo, apertando o botão INACTIVE do PX2612 para este CP, ou comando equivalente nos diagnósticos da redundância. Desta forma, evita-se que o outro CP desligue este CP e assuma como Ativo.

ATENÇÃO:
Quando o CP Ativo sai do modo Run e vai para o estado Não-Configurado, se o outro CP foi esquecido em estado Reserva, o outro CP assumirá como Ativo e desligará o CP que passou de Ativo para Não-Configurado. Neste caso, portanto, a carga off-line não poderá ser completada porque o CP conectado ao MasterTool foi desligado.

Quando a carga off-line terminar, é possível reiniciar a execução do programa no CP onde a aplicação foi carregada (colocar no modo Run novamente). Depois de alguns segundos, este CP reassume o estado Ativo.

Depois que este CP reassume o estado Ativo, pode-se tirar o outro CP do estado Inativo, por exemplo, apertando o botão STAND-BY do PX2612 para este CP. Isso provocará a transição deste CP para o estado Não-Configurado. Este CP permanecerá no estado Não-Configurado até que o processo automático de sincronização de projetos termine. Depois disso, este CP passa para o estado Inicializando e depois voltará para o estado Reserva.

Planejamento Prévio para Modificações Off-Line sem Interrupção do Controle do Processo

A sub-seção seguinte – Planejamento Prévio para Modificações a Quente em Redes PROFIBUS Redundantes – descreve um procedimento muito importante que permite fazer cargas off-line de modificações sem interromper o controle do processo. Embora este procedimento não se aplique a quaisquer modificações que demandem carga off-line, ele se aplica àquelas modificações realizadas com maior frequência.

No entanto, para que o procedimento seja aplicável, os projetos devem ser desenvolvidos com algum planejamento prévio, particularmente para modificações que afetam a rede PROFIBUS. As seguintes subseções descrevem tais planejamentos prévios para modificações que afetam a rede PROFIBUS, e também para outras modificações.

Planejamento Prévio para Modificações a Quente em Redes PROFIBUS Redundantes

Entre as modificações que afetam uma rede PROFIBUS e demandam carga off-line, as seguintes são suportadas pelo procedimento que permite fazer cargas off-line de modificações sem interromper o controle do processo, desde que a rede PROFIBUS seja redundante:

- Inserir uma nova rede PROFIBUS
- Inserir uma nova remota da série Ponto
- Inserir um novo módulo de E/S em uma remota da série Ponto
- Modificar parâmetros em remotas da série Ponto ou em módulos de E/S de remotas da Série Ponto

ATENÇÃO: É possível inserir remotas não-redundantes abaixo de uma rede PROFIBUS redundante utilizando o módulo AL-2433 (ProfiSwitch), conforme exemplo mostrado na Figura 6-1. No entanto, tais remotas não-redundantes sofrerão descontinuidades (desligamento de saídas) quando forem carregadas modificações off-line.

A seguir, descreve-se as etapas de um planejamento que deve iniciar no momento da criação de uma nova rede PROFIBUS redundante.

Etapa 1 – Planejar Expansão Futura das Remotas Inseridas na Versão Inicial da Rede PROFIBUS

Em primeiro lugar, deve-se fazer uma lista dos módulos de E/S que constituirão cada remota PROFIBUS redundante da Série Ponto, na versão inicial da rede PROFIBUS. Na lista, deve constar a posição onde cada módulo de E/S será inserido no bastidor da remota.

A seguir, deve-se planejar como cada uma destas remotas poderá ser expandida no futuro. Para tanto, deve-se fazer uma lista complementar com módulos de E/S que poderão ser inseridos futuramente. Na lista complementar, deve constar a posição onde cada módulo de E/S será inserido futuramente no bastidor de cada remota.

ATENÇÃO:
Na construção física destas remotas (armários elétricos), é altamente recomendável inserir bases compatíveis com os módulos de E/S futuros nas respectivas posições. Desta maneira, quando for necessário inserir o módulo de E/S nesta remota, não será necessário desligar a mesma para inserir a base. Caso este detalhe não seja observado, será necessário desligar esta remota específica, pois não é possível inserir uma base à quente em uma remota. Observa-se que a parada de uma remota específica em alguns casos pode ser tolerável, mas em outros casos pode não ser.

ATENÇÃO:
Deve-se colocar os módulos de E/S originais nas primeiras posições do bastidor da remota, e os módulos de E/S futuros nas últimas posições do bastidor da remota.

ATENÇÃO:
Considerar as limitações das remotas redundantes da Série Ponto ao elaborar esta lista, conforme o Manual de Utilização Cabeça PROFIBUS PO5063V1 e Cabeça Redundante PROFIBUS PO5063V5, e o Manual de Utilização Cabeça PROFIBUS PO5064 e Cabeça Redundante PROFIBUS PO5065. Existem limites quanto ao número de módulos por remota, número de bytes de E/S por remota, consumo de corrente por fonte, etc. Estes limites podem ser verificados automaticamente utilizando o ProPonto. Para maiores informações, consulte o Manual de Utilização MT6000 MasterTool ProPonto – MU299040.

Etapa 2 – Inserir a Versão Inicial da Rede PROFIBUS Redundante no Projeto

Para inserir a versão inicial de rede PROFIBUS redundante no Projeto, inicialmente deve-se inserir os dois módulos NX5001 redundantes no bastidor, ou utilizar aqueles já inseridos pelo Wizard da redundância.

Em seguida, deve-se inserir cada uma das remotas na árvore de dispositivos abaixo do primeiro NX5001 do par redundante na árvore de dispositivos, bem como os módulos de E/S abaixo de cada remota.

Quanto aos módulos de E/S inseridos, existem duas categorias que devem ser tratadas de maneira diferente:

- Aqueles que fazem parte da versão inicial da rede PROFIBUS, e serão instalados imediatamente
- Aqueles que serão utilizados para expansão futura

No caso daqueles que fazem parte da versão inicial da rede PROFIBUS, deve-se inserir o próprio módulo na árvore de dispositivos, na posição planejada da remota correspondente.

No caso daqueles que serão utilizados para expansão futura, deve-se inserir um módulo virtual correspondente na posição planejada. Um módulo virtual simula um módulo real e precisa alocar o mesmo número de bytes de E/S que o módulo real. A inserção de um módulo virtual no lugar de um módulo real impede que diagnósticos de ausência do módulo real sejam produzidos.

A Tabela 6-4 exemplifica alguns módulos reais e seus respectivos módulos virtuais correspondentes:

Módulo Real	Módulo Virtual Correspondente
PO1000	PO9999 – 2 bytes Input
PO1001	PO9999 – 2 bytes Input
PO1002	PO9999 – 2 bytes Input
PO1003	PO9999 – 2 bytes Input
PO2020	PO9999 – 2 bytes Output
PO2022	PO9999 – 2 bytes Output

Tabela 6-4. Módulos Virtuais Correspondentes aos Módulos Reais

Etapa 3 – Alocar Áreas de Variáveis %I e %Q para a Rede PROFIBUS considerando Expansão para Remotas Futuras

Na medida que os NX5001, remotas e módulos de E/S foram sendo inseridos na árvore de dispositivos na etapa 2 anterior, foram alocadas variáveis %I e %Q em três áreas diferentes:

- Área de variáveis %I para entradas
- Área de variáveis %Q para saídas
- Área de variáveis %Q para diagnósticos

O MasterTool executa a alocação de cada uma destas três áreas de variáveis de maneira contígua, sem deixar intervalos nas alocações.

Deve-se planejar o endereço inicial e final de cada uma destas três áreas, considerando as remotas inicialmente instaladas na rede (ver etapas 1 e 2 anteriores), mas também remotas que poderão ser inseridas no futuro nesta mesma rede PROFIBUS.

Ao definir o endereço inicial de cada área, é importante reservar expansão para o dispositivo que aloca endereços imediatamente antes do início desta área. Por outro lado, ao definir o endereço final de cada área, é importante reservar expansão para esta rede PROFIBUS.

A seguir, mostra-se um exemplo de tal planejamento, para a área de variáveis %I de entrada:

- Rede PROFIBUS 1:
  - %IB0 ... %IB499 (endereços alocados para remotas já instaladas)
  - %IB500 ... %IB999 (endereços alocados para remotas futuras)
- Rede PROFIBUS 2:
  - %IB1000 ... %IB1499 (endereços alocados para remotas já instaladas)
  - %IB1500 ... %IB1999 (endereços alocados para remotas futuras)
- Servidor MODBUS TCP:
  - %IB2000 ... %IB2999 (endereços alocados para mapeamentos atuais)
  - %IB3000 ... %IB3999 (endereços alocados para mapeamentos futuros)

Para as outras duas áreas (%Q de saída, %Q de diagnósticos) poderia-se mostrar exemplos similares de planejamentos.

É possível prever o tamanho das áreas inicialmente alocadas e de expansão futura utilizando a Tabela 6-5, que indica as quantidades de bytes alocados em cada uma das três áreas, para cada módulo:

Módulo	%I entradas (bytes)	%Q saídas (bytes)	%Q diagnósticos (bytes)
NX5001	4	2	86
PO5063V5	0	0	25
PO5065	0	0	25
PO9100 (um por remota)	2	2	10
PO1000	2	0	10
PO2020	0	2	10
PO9999 – 2 bytes Output	0	2	10
PO9999 – 2 bytes Input	2	0	10

Tabela 6-5. Alocação de Variáveis %I e %Q para Módulos da Rede PROFIBUS

Nota:

Alocação de Variáveis: Mais informações sobre o tamanho e o tipo de memória alocado para cada módulo podem ser encontradas no Manual de utilização Mestre PROFIBUS DP NX5001.

Depois de executar o planejamento das três áreas (endereços inicial e final de cada área), deve-se introduzir os endereços iniciais no projeto iniciado na etapa 2.

Em primeiro lugar, deve-se modificar o parâmetro "Diagnostic Offset in %Q Area" do primeiro módulo NX5001. Deve-se utilizar o endereço inicial planejado para a área de variáveis %Q para diagnósticos.

Em segundo lugar, deve-se procurar o primeiro módulo de E/S da rede, a partir dos NX5001, que aloque variáveis %I para entradas. Ao encontrá-lo, deve-se alterar o parâmetro "Address" correspondente.

Em terceiro lugar, deve-se procurar o primeiro módulo de E/S da rede, a partir dos NX5001, que aloque variáveis %Q para saídas. Ao encontrá-lo, deve-se alterar o parâmetro "Address" correspondente.

ATENÇÃO: Neste momento, é aconselhável verificar a faixa de endereços alocados para as três áreas de variáveis, verificando se os endereços finais de cada área estão dentro da faixa planejada, e se existe uma boa expansão livre para novas remotas a serem inseridas futuramente.

Planejamento Prévio para Outras Modificações

Existem outras modificações, que embora não afetem uma rede PROFIBUS, também demandam carga off-line. A seguir, são mostrados alguns exemplos de modificações deste tipo suportadas pelo procedimento que permite fazer cargas off-line de modificações sem interromper o controle do processo:

- Inserção de módulos NX5000 (Ethernet)
- Inserção de I/O drivers de comunicação Ethernet ou Serial
- Inserção de novos mapeamentos em I/O drivers de comunicação Ethernet ou Serial
- Modificar o período da MainTask

Algumas modificações simples, tais como modificar o período da MainTask, não exigem nenhum planejamento prévio.

Por outro lado, as demais modificações exemplificadas anteriormente nesta seção envolvem a alocação de variáveis de representação direta %I e %Q para diagnósticos, entradas e saídas, de maneira similar ao que se discutiu na etapa 3 do planejamento prévio para modificações a quente que afetam a rede PROFIBUS (ver seção Etapa 3 – Alocar Áreas de Variáveis %I e %Q para a Rede PROFIBUS considerando Expansão para Remotas Futuras).

Desta forma, ao inserir um módulo NX5000, ou ao inserir um I/O driver Ethernet ou Serial, deve-se planejar a alocação das 3 seguintes áreas para o dispositivo inserido:

- Área de variáveis %I para entradas
- Área de variáveis %Q para saídas
- Área de variáveis %Q para diagnósticos

A seção Etapa 3 – Alocar Áreas de Variáveis %I e %Q para a Rede PROFIBUS considerando Expansão para Remotas Futuras mostra um exemplo de alocação conjunta destas áreas, envolvendo redes PROFIBUS e um I/O driver (Servidor MODBUS TCP).

Incompatibilidade de Aplicações

Caso as áreas a serem utilizadas futuramente não sejam planejadas corretamente, as áreas de memória redundante podem ter que ser alteradas, gerando, assim, uma incompatibilidade entre as aplicações. Isto irá implicar em apenas um CP manter-se no estado Ativo, com o outro CP permanecendo Inativo, sem a possibilidade de sincronizar dados redundantes ou a aplicação entre os dois CPs.

Esta incompatibilidade é informada pelo diagnóstico da redundância:
DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.bApplicationIncompatible.

Este diagnóstico é ativo quando o download de uma nova aplicação é feito para um dos CPs, geralmente o Não-Ativo, com uma das seguintes alterações:

- Modificação das áreas de dados redundantes, configuradas nos parâmetros do módulo NX4010
- Alteração (criar ou remover) de variáveis simbólicas redundantes, declaradas em POUs ou GVLs redundantes

É importante observar que, para a alteração de variáveis simbólicas redundantes, o problema de incompatibilidade ocorre somente com um novo download para um dos CPs. Caso a modificação das variáveis simbólicas redundantes, e todas as outras alterações feitas no projeto, se enquadrem no grupo de Modificações que Permitem Carga On-Line, é possível fazer uma Carga On-Line de Modificações e não gerar nenhuma incompatibilidade entre as aplicações dos CPs.

Atualizações de Projeto Devido a Atualizações do MasterTool IEC XE

A ferramenta de programação MasterTool IEC XE passa por um constante processo de melhoria, aprimorando suas características e adicionando novas. Quando é necessária a atualização da ferramenta em um sistema redundante o projeto utilizado precisa ser atualizado. Esta atualização é realizada através do menu Projeto/Atualização de Projeto disponível na ferramenta. Após atualizar o projeto o procedimento de carga Off-Line sem Interrupção do Controle do Processo pode ser realizado.

Atualização de Projetos de Versões Inferiores a 2.00 para versão 2.00 ou Superior

Dentre as trocas entre versões do software MasterTool IEC XE existe um caso especial que deve ser planejado com maior cautela para evitar a parada do processo. A atualização de projeto criado com versão inferior a 2.00 do MasterTool IEC XE para a versão 2.00 ou superior causa uma reconfiguração na área alocada para objeto Variáveis Persistentes. Esta reconfiguração foi implementada visando uma otimização na alocação desta área. Contudo, se este objeto estiver presente e marcado como redundante em um projeto, esta reorganização não permitirá que os dados sejam sincronizados entre os dois projetos, colocando sempre um dos Half-Clusters em estado Inoperante.

Desta forma, se esta situação ocorrer, o software da UCP NX3030 consegue detectar a situação e para a sincronização de dados do objeto Variáveis Persistentes até que os projetos nos dois Half-Clusters sejam iguais e por conseguinte estejam utilizando um projeto com a versão atualizada do MasterTool IEC XE. Esta situação não irá parar o controle do processo, porém se não for seguida uma sequência correta de atualização, os dados do objeto Variáveis Persistentes podem ser reinicializados.

Nestes casos a seguinte sequência de Carga Off-Line deve ser realizada:

- Alterar o projeto do Half-Cluster em estado Ativo desmarcando o objeto PersistentVars dentro do objeto Redundancy Configuration. Esta carga deve ser realizada como uma carga On-Line e para que possa ser feita se faz necessária mais uma alteração no projeto, por exemplo, declarando uma variável dentro da POU NonSkippedPrg
- Ao final da Carga On-Line será necessário executar o comando Criar Aplicação de Inicialização quando estiver On-line no CP que se encontra no estado Ativo. Isto é necessário para que a aplicação seja sincronizada com o Half-Cluster que passou para o estado Não Configurado após a carga
- Atualizar o projeto da versão inferior a 2.00 para versão 2.00 ou superior através do menu Projeto\Atualização de Projeto do MasterTool IEC XE
- Desabilitar o sincronismo de projeto através do menu Comunicação\Configuração de Redundância
- Executar a carga de projeto atualizado no Half-Cluster que se encontra no estado Reserva. Aparecerá uma mensagem indicando a reorganização das áreas de memória do objeto PersistentVars. O procedimento deve continuar e ao final da carga de projeto o Half-Cluster permanecerá em STOP com estado de redundância como Não Configurado

- Passar a UCP para RUN. O Half-Cluster irá transicionar para estado Inicializando e depois para o estado Reserva. O Half-Cluster irá sincronizar seus dados com o outro que está no estado Ativo
- Os dados do objeto PersistentVars devem ser copiados manualmente do Half-Cluster Ativo para o Reserva ou deve se utilizar o recurso de receitas para copiar os dados de um Half-Cluster para o outro
- Colocar o Half-Cluster do estado Ativo no estado Reserva. Com esta ação o estado do outro Half-Cluster passará para Ativo
- Habilitar o sincronismo de projeto através do menu Comunicação\Configuração de Redundância. Após este processo o Half-Cluster que estava no estado Reserva irá para o estado Não Configurado e receberá o projeto do Half-Cluster em estado Ativo. Ao final deste processo o estado do Half-Cluster mudará para Inicializando e por fim devolta a Reserva
- Alterar o projeto do Half-Cluster em estado Ativo marcando o objeto PersistentVars dentro do objeto Redundancy Configuration. Esta carga deve ser realizada como uma carga On-Line e para que possa ser feita se faz necessária mais uma alteração no projeto, por exemplo, removendo uma variável dentro da POU NonSkippedPrg (por exemplo a declarada no início deste procedimento)
- Após este processo o Half-Cluster que estava no estado Reserva irá para o estado Não Configurado e receberá o projeto do Half-Cluster em estado Ativo. Ao final deste processo o estado do Half-Cluster mudará para Inicializando e por fim devolta a Reserva

Explorando a Redundância para Carga Off-Line de Modificações sem Interrupção do Controle do Processo

Na seção Carga de Modificações Off-Line e On-Line, informou-se que algumas modificações demandam carga off-line no CP onde tais modificações devem ser carregadas.

Nestes casos, o usuário tem a opção de interromper o controle do processo, conforme procedimento definido na seção Carga Off-Line de Modificações com Interrupção do Controle do Processo. Para tanto, geralmente é necessário programar previamente uma parada do processo, o que nem sempre é possível no momento em que é necessária uma modificação urgente.

Graças à redundância dos CPs e, em alguns casos, graças à redundância da rede PROFIBUS, é possível realizar cargas off-line sem interromper o controle do processo para a maior parte das modificações usualmente necessárias. Para atingir este objetivo, é necessário seguir atentamente um procedimento, cujas etapas são descritas nas subseções seguintes.

Etapa 1 – Verificar Atendimento de Requisitos Básicos

Para que sejam possíveis cargas off-line sem interromper o controle do processo, os seguintes requisitos básicos devem ser atendidos:

- O projeto original deve ter sido elaborado em conformidade com as recomendações dadas na seção Planejamento Prévio para Modificações Off-Line sem Interrupção do Controle do Processo
- O CP deve ser redundante
- Caso a modificação afete uma rede PROFIBUS, é necessário que esta rede PROFIBUS seja redundante. Tais modificações podem ser:
    - Inserção de novas remotas
    - Inserção de módulos de E/S em remotas existentes, em posições previamente reservadas por módulos virtuais correspondentes. Para que a remota não tenha de ser desligada, além disso, deve-se ter uma base compatível com o novo módulo de E/S na posição reservada para o mesmo
    - Modificação de parâmetros em remotas ou módulos de E/S existentes
- Os projetos de ambos CPs devem estar sincronizados (iguais), e os serviços Sincronização de Dados Redundantes e Sincronização da Lista de Forçamentos Redundantes devem estar funcionando corretamente sem nenhum diagnóstico de falha. Estas condições estão satisfeitas quando existe um CP em estado Ativo e outro em estado Reserva. Caso o CP Não-Ativo não esteja em estado Reserva, pode-se observar seus diagnósticos:

- DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.bRedDataSync = TRUE, indica sucesso no serviço Sincronização de Dados Redundantes
- DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.bRedForceSync = TRUE, indica sucesso no serviço Sincronização da Lista de Forçamentos Redundantes
- DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.dwApplicationCRC = DG_NX4010.tRedundancy.RedDgnRem.dwApplicationCRC, indica que os projetos são iguais nos 2 CPs

Etapa 2 – Não Carregar em Conjunto Modificações que podem ser Carregadas On-Line

Modificações que podem ser carregadas on-line não devem ser carregadas junto com modificações que devem ser carregadas off-line sem interrupção do controle do processo. Sempre que for necessário fazer estes dois tipos de modificações, deve-se executá-las e carregá-las separadamente.

Para que o presente procedimento tenha sucesso, é absolutamente necessário que as modificações realizadas não alterem a estrutura das variáveis redundantes trocadas entre CP Ativo e Não-Ativo, através dos serviços Sincronização de Dados Redundantes e Sincronização da Lista de Forçamentos Redundantes. Estes dois serviços devem continuar funcionando corretamente mesmo enquanto houver diferenças temporárias entre os projetos dos dois CPs.

As modificações que devem ser carregadas off-line, e suportadas por este procedimento, não afetam a estrutura das variáveis redundantes.

No entanto, algumas modificações que podem ser carregadas on-line podem alterar a estrutura das variáveis redundantes, por exemplo:

- Inserção de variáveis simbólicas (redundantes ou não), dentro de uma POU ou GVL existente, ou em uma nova POU ou GVL
- Remoção de variáveis simbólicas (redundantes ou não), dentro de uma POU ou GVL existente. A remoção de uma POU ou GVL também pode implicar na remoção de variáveis simbólicas
- Modificação de tamanho/estrutura de variáveis simbólicas (redundantes ou não) em uma POU ou GVL existente

Etapa 3 – Backup do Projeto Anterior

Antes de editar as modificações que devem ser carregadas off-line sem interromper o processo, por segurança, deve-se realizar um backup da versão anterior do projeto. Poderá ser necessário reinstalar a versão anterior caso algum erro seja cometido durante a execução deste procedimento.

ATENÇÃO: A recomendação de ter um backup de todas as versões que foram carregadas nos CPs não deve ser seguida somente para este procedimento específico. Deve ser uma prática usual.

Etapa 4 – Cuidados ao Editar as Modificações Carregadas Off-Line

As modificações carregadas off-line através deste procedimento geralmente são as seguintes:

- Inserção de novos dispositivos na árvore de dispositivos
- Alteração de propriedades ou parâmetros em dispositivos existentes na árvore de dispositivos

Tais dispositivos normalmente são os seguintes:

- Módulos tais como mestres PROFIBUS (NX5001) ou módulos Ethernet (NX5000)
- Remotas PROFIBUS da Série Ponto
- Módulos de E/S dentro de remotas PROFIBUS da Série Ponto
- I/O drivers de comunicação MODBUS
- Mapeamentos de drivers de comunicação MODBUS

Os seguintes cuidados devem ser tomados ao editar estas modificações no projeto:

- Se um mesmo dispositivo existia na versão anterior do projeto e continua existindo na versão modificada, as variáveis %I e %Q alocadas para o mesmo devem continuar sendo as mesmas (comandos, diagnósticos, entradas e saídas). Deve-se ter cuidado para que as modificações inseridas no projeto não alterem tais alocações
- Se um dispositivo foi inserido na versão modificada do projeto, as variáveis %I e %Q alocadas para o mesmo devem estar desalocadas na versão anterior do projeto (comandos, diagnósticos, entradas e saídas)

Etapa 5 – Desabilitar Sincronismo de Projetos no CP Não-Ativo

Nos procedimentos descritos nas seções Carga On-Line de Modificações e Carga Off-Line de Modificações com Interrupção do Controle do Processo, o projeto é sincronizado automaticamente, do CP Ativo para o CP Não-Ativo.

No entanto, durante o procedimento de carga off-line sem interrupção do controle do processo, o sincronismo de projetos deve ser temporariamente desabilitado.

Este processo é explicado na seção Desabilitação da Sincronização de Projetos, e a desabilitação deve ser executada no CP Não-Ativo.

Etapa 6 – Executar as Modificações Físicas

Neste momento, pode-se executar as modificações físicas, tais como:

- Instalar um novo módulo NX5000. Isto pode ser feito inserindo o módulo a quente no rack de cada half-cluster, e depois conectando-o à rede Ethernet
- Instalar uma nova rede PROFIBUS redundante. Os módulos NX5001 podem ser instalados a quente no rack de cada half-cluster. Em seguida, a rede PROFIBUS redundante pode ser conectada aos mesmos
- Instalar uma nova remota redundante da Série Ponto. Neste caso, deve-se instalar uma cabeça remota de cada vez, por exemplo, primeiro na rede B, depois na rede A:
    - Para instalar a cabeça na rede B, pode ser necessário abrir o cabo ou a terminação, e desta maneira perturbar a comunicação com as demais cabeças já instaladas na rede B. Antes de fazer isso, deve-se colocar todas as cabeças ativas operando na rede A, e as cabeças reservas operando na rede B
    - Para instalar a cabeça na rede A, pode ser necessário abrir o cabo ou a terminação, e desta maneira perturbar a comunicação com as demais cabeças já instaladas na rede A. Antes de fazer isso, deve-se colocar todas as cabeças ativas operando na rede B, e as cabeças reservas operando na rede A
- Instalar um módulo de E/S em uma base previamente reservada para o mesmo, em uma remota existente

Etapa 7 – Carregar as Modificações Off-Line no CP Não-Ativo

Em primeiro lugar, o MasterTool deve estar conectado ao CP Não-Ativo (ver seção Conexão do MasterTool com uma UCP NX3030 de um CP Redundante).

A seguir, deve-se carregar as modificações off-line. Ao carregá-las, a aplicação do CP Não-Ativo será automaticamente interrompida (sairá do modo Run).

Etapa 8 – Voltar o CP Não-Ativo ao Modo Run para que volte ao Estado Reserva

Terminada a carga off-line, pode-se voltar o CP Não-Ativo para o modo Run.

Alguns segundos depois, o CP Não-Ativo deve assumir o estado Reserva.

Caso o CP não assuma o estado Reserva, os seguintes problemas podem ter causado este efeito:

- As modificações realizadas acabaram introduzindo modificações na estrutura das variáveis redundantes, o que impede a execução correta do serviço Sincronização de Dados Redundantes.

Isto pode ser verificado através do diagnóstico DG_NX4010.tRedundancy.RedDgnLoc. sGeneral_Diag.bRedDataSync (0 = falha) no CP Não-Ativo. Neste caso, deve-se desfazer as modificações, recuperando o backup do projeto anterior, e iniciando novamente este procedimento

- Outros problemas podem, eventualmente, impedir a transição para o estado Reserva, embora isto não seja esperado. Neste caso, deve-se observar os diagnósticos e log da redundância

Caso o CP tenha assumido o estado Reserva, é aconselhável verificar se os projetos estão diferentes entre CP Ativo e CP Não-Ativo. Isto pode ser feito comparando os diagnósticos DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.dwApplicationCRC e DG_NX4010.tRedundancy.RedDgnRem.dwApplicationCRC no CP Não-Ativo (os CRCs devem estar diferentes).

Caso os projetos estejam iguais nos 2 CPs, é possível que a desabilitação de sincronismo de projetos (etapa 5) não tenha sido realizada corretamente. Isto pode ser verificado através do diagnóstico DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag. bProjectSyncDisable, que deveria valer 1 no CP Não-Ativo. Se isto não ocorre, deve-se voltar à etapa 5.

Etapa 9 – Executar Switchover entre CPs Ativo e Reserva

Deve-se executar switchover entre os CPs, por exemplo, apertando o botão STAND-BY do CP Ativo, ou executando o comando equivalente. O CP Reserva, que possui o projeto novo com as modificações, assume como Ativo. O CP Ativo, que possui o projeto antigo, assume como Reserva.

Etapa 10 – Habilitar Sincronismo de Projetos no CP Ativo

Na etapa 5, o sincronismo de projetos foi desabilitado no CP que estava em estado Não-Ativo. Observa-se que agora este CP está em estado Ativo.

Nesta etapa, deve-se habilitar novamente o sincronismo de projetos neste CP. A tela e metodologia utilizada para isto é a mesma descrita na seção Desabilitação da Sincronização de Projetos, bastando desta vez selecionar a opção “Habilitar” da combo-box. Desta vez, o MasterTool deve estar conectado ao CP Ativo (ver seção Conexão do MasterTool com uma UCP NX3030 de um CP Redundante).

Depois de habilitar o sincronismo de projetos no CP Ativo, o usuário deve conferir se este comando teve sucesso, verificando se DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag. bProjectSyncDisable = 0, no CP Ativo.

Assim que o sincronismo de projetos for novamente habilitado, a seguinte sequência de ações é esperada:

- O CP Não-Ativo (em estado Reserva), que já sabia da diferença entre os projetos, sai do estado Reserva para o estado Não-Configurado
- O projeto modificado (novo) é copiado do CP Ativo para o outro CP, temporariamente em estado Não-Configurado
- Assim que os projetos estiverem novamente sincronizados, o CP Não-Configurado passa para o estado Inicializando e, depois, espera-se que volte ao estado Reserva

Etapa 11 – Reorganização Opcional de CP e Redes PROFIBUS em Estado Ativo

No final do procedimento, por questões de padronização ou organização, o usuário poderá fazer um switchover para que o CPA assuma como Ativo, e para que todas as cabeças remotas PROFIBUS em estado ativo estejam na rede A.

Manutenção

Troca a Quente de Módulos em um CP Redundante

Em caso de falha em algum módulo de um dos CPs (CPA ou CPB), pode ser necessária uma troca do módulo, sem interromper o controle do processo. Para isso, os seguintes passos devem ser seguidos:

- Verificar se o half-cluster que não será modificado está em estado Ativo ou em estado Reserva, podendo assumir o controle do processo
- Colocar o half-cluster em que será efetuada a troca do módulo em estado Inativo, através do Painel de Controle de Redundância PX2612 ou dos Comandos da Redundância
- Efetuar as trocas necessárias no half-cluster Inativo, conforme indicado no capítulo Configuração da UCP - Parâmetros Gerais - Como realizar a Troca a Quente
- Colocar o half-cluster novamente em estado Reserva ou Ativo, conforme o necessário

Mensagens de Advertência do MasterTool

Quando o MasterTool está com um projeto redundante aberto, ou quando está conectado a uma UCP NX3030 identificada como CPA ou CPB, algumas mensagens de advertência especiais poderão ocorrer, conforme descreve-se nas próximas subseções.

Bloqueio de Carga de um Projeto Redundante ou Não-Redundante

O MasterTool não permitirá a carga de um projeto redundante, a menos que a UCP seja NX3030, e que esta UCP esteja identificada como CPA ou CPB (ver seção Identificação de uma UCP NX3030).

O MasterTool também não permitirá a carga de um projeto não-redundante em uma UCP NX3030 identificada como CPA ou CPB.

Caso ocorra uma tentativa de executar alguma destas ações ilegais, o MasterTool advertirá com uma mensagem apropriada.

Alertas antes de Comandos que Podem Parar o CP Ativo

Alguns comandos, como os seguintes, podem parar um CP:

- Carga offline após Comunicação / Login
- Depurar / Parar
- Depurar / Novo Breakpoint
- Comunicação / Reset (quente, frio, origem)

Caso o MasterTool esteja logado ao CP ativo e ocorra uma tentativa de execução de um destes comandos, antes de enviá-lo, o MasterTool envia a seguinte mensagem e aguarda autorização para enviar o comando:

"Caso o outro CP estiver no estado Reserva, ele irá assumir como Ativo e desligará este CP. Caso contrário, isto não irá acontecer, mas o processo automatizado não será mais controlado."

Alerta antes de Logar-se ao CP Não-Ativo

Em circunstâncias normais, não é usual o MasterTool logar-se ao CP Não-Ativo, portanto, quando houver uma tentativa de execução de um comando deste tipo, o MasterTool emite o seguinte aviso:

"Você está executando um comando de Login em um CP Não-Ativo e isso não é usual. Você tem certeza que deseja executar esse comando?"

Por outro lado, há circunstâncias (não muito usuais) em que é necessário logar-se ao CP Não-Ativo, e nestes casos o usuário deve autorizar o login. Tais circunstâncias podem ocorrer, por exemplo:

- Para configurações iniciais, conforme descrito na seção Carga Inicial de um Projeto Redundante

- Para carregar off-line um projeto diferente no CP Não-Ativo, conforme descrito na seção Explorando a Redundância para Carga Off-Line de Modificações sem Interrupção do Controle do Processo
- Para monitorar ou forçar variáveis não-redundantes no CP Não-Ativo

Diagnósticos da Redundância no Visor Gráfico da UCP NX3030

Diversos diagnósticos relativos à redundância podem ser observados no visor da UCP NX3030.

Estado de Redundância do CP

O estado de redundância do CP, descritos em Estados de um CP Redundante, é visto nos três caracteres iniciais da segunda linha da tela principal, conforme mostra o capítulo Visor Gráfico. A tela do visor é apresentada na inicialização, e volta a ser apresentada alguns segundos depois de terminada a navegação (sem apertar o botão da UCP NX3030).

Telas abaixo do Menu REDUNDANCIA

Existe um menu denominado REDUNDANCIA, abaixo do qual existem diversas telas. A descrição e o acesso às telas de redundância estão disponíveis no capítulo Configuração –Menu Informativo e de Configuração da UCP.

Estrutura de Diagnósticos da Redundância

A área de diagnósticos do módulo NX4010 é mapeada sobre variáveis de representação direta %Q, e definida simbolicamente através da diretiva AT, na GLV Diagnostics.

Essa seção é dividida em duas partes:

- DG_NX4010.tGeneral: Diagnósticos gerais de operação do módulo NX4010. Os mesmos são descritos nas Características Técnicas do Módulo de Link de Redundância – CT114900
- DG_NX4010.tRedundancy: Diagnósticos específicos da redundância, os quais serão descritos no decorrer do capítulo. Esse item é dividido em outras 6 estruturas de dados:
    - RedDgnLoc: Contém diagnósticos de redundância do CP local (conectado), como por exemplo, o estado da redundância do CP. Essa seção é descrita em Diagnósticos da Redundância
    - RedDgnRem: É uma cópia de RedDgnLoc do outro CP, recebida via canais de sincronismo NETA / NETB. Desta forma, o CP local tem acesso aos diagnósticos do CP remoto. Essa seção é descrita em Diagnósticos da Redundância
    - RedCmdLoc: Contém comandos de redundância gerados neste CP (local), por exemplo, através de escritas a partir de um sistema SCADA, ou gerados em POUs deste CP (ex: ActivePrg ou NonSkippedPrg). Essa seção é descrita em Comandos da Redundância
    - RedCmdRem: Trata-se de uma cópia de RedCmdLoc do outro CP (remoto), recebida via canais de sincronismo NETA / NETB. Essa seção é descrita em Comandos da Redundância
    - RedUsrLoc: Utilizado para que o usuário troque informações entre CPA e CPB. Essa seção é descrita em Informações do Usuário Trocados entre CPA e CPB
    - RedUsrRem: Utilizado para que o usuário troque informações entre CPA e CPB. Essa seção é descrita em Informações do Usuário Trocados entre CPA e CPB

É importante ressaltar que as estruturas de diagnósticos da redundância do CP remoto são atualizadas somente quando ocorre, com sucesso, uma sincronização de dados. Portanto, enquanto uma nova sincronização não ocorrer, os diagnósticos irão permanecer com o valor lido na última troca de dados.

Além disso, as estruturas do CP remoto são somente para leitura, isto é, valores escritos nestas estruturas serão sobre-escritos, sem serem considerados, na próxima sincronização. Sendo assim, não é possível utilizar a estrutura RedCmdRem para executar um comando no CP remoto. A estrutura utilizada para executar comandos deve ser sempre RedCmdLoc.

Diagnósticos da Redundância

Os diagnósticos da Redundância podem ter várias utilidades. Entre elas:

- Podem ser consultados para verificar a existência de algum problema que precisa ser sanado
- Toda vez que ocorrem variações nos mesmos, tais variações são inseridas como eventos nos logs de usuário. Consultando a sequência histórica de tais eventos pode-se descobrir, por exemplo, a causa de um switchover
- Podem ser referenciados na aplicação do usuário (ActivePrg ou NonSkippedPrg). Por exemplo, pode-se testar o estado do CP e, caso não seja Ativo, desabilitar um I/O driver mestre serial MODBUS RTU, em NonSkippedPrg

ATENÇÃO:
O diagnóstico DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.bExchangeSync (definido logo a seguir) deve ser testado para verificar se a estrutura de dados RedDgnRem foi lida com sucesso do CP remoto no último ciclo da MainTask. Caso o valor deste diagnóstico seja FALSE, isso significa que a estrutura de dados RedDgnRem não foi lida com sucesso do CP remoto e, portanto, os valores de RedDgnRem podem ser inválidos ou obsoletos.

Como RedDgnRem é uma cópia de RedDgnLoc do outro CP, as duas estruturas têm o mesmo formato. Estas ainda são divididas em quatro sub-estruturas:

- sGeneral_Diag: Diagnósticos gerais da redundância
- sNETA_Diag: Diagnósticos do canal de sincronismo NETA
- sNETB_Diag: Diagnósticos do canal de sincronismo NETB
- sNET_Stat: Estatísticas comuns para os canais de sincronismo NETA e NETB, para contagem de sucessos e falhas dos serviços de sincronização

A sub-estrutura “sGeneral_Diag” possui os seguintes campos para diagnósticos gerais da redundância:

Variável de Representação Direta		Variável AT DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.*	Descrição
Variável	**Bit**		
%QB(n+4)	0	bConfigDone	TRUE – O processo de configuração, executado no estado Não-Configurado, terminou.
			FALSE – O processo de configuração, executado no estado Não-Configurado, ainda não terminou ou não foi executado.
	1	bConfigError	TRUE – O processo de configuração, executado no estado Não-Configurado, terminou com erros. Trata-se de um erro de sistema, normalmente não esperado. Entre em contato com o suporte da Altus para reportá-lo. Informe também o valor do diagnóstico ConfigErrorCode para o suporte da Altus.
			FALSE – O processo de configuração ocorreu com sucesso ou não foi realizado.
	2	bTooManyRedAreas	TRUE – O número de áreas redundantes excedeu o número máximo permitido. Trata-se de um erro de sistema, normalmente não esperado. Entre em contato com o suporte da Altus para reportá-lo.
			FALSE – O número de áreas redundantes está dentro do esperado.
	3	bTemporaryBufferTooSmall	TRUE – Estrutura de dados intermediária com tamanho insuficiente. Trata-se de um erro de sistema, normalmente não esperado. Entre em contato com o suporte da Altus para reportá-lo.
			FALSE – O tamanho da estrutura de dados está dentro do esperado.
	4	bExchangeSync	TRUE – O serviço de sincronização Troca de Diagnósticos e Comandos foi executado com

			sucesso neste ciclo da MainTask.
			FALSE – A estrutura RedDgnRem tem valores obsoletos ou inválidos, pois não foi lida do outro CP (remota) neste ciclo da MainTask.
	5	bRedDataSync	TRUE – O serviço Sincronização de Dados Redundantes foi executado com sucesso neste ciclo da MainTask.
			FALSE – O serviço Sincronização de Dados Redundantes não foi executado com sucesso neste ciclo da MainTask.
	6	bRedForceSync	TRUE – O serviço Sincronização da Lista de Forçamentos Redundantes foi executado com sucesso neste ciclo da MainTask.
			FALSE – O serviço Sincronização da Lista de Forçamentos Redundantes não foi executado com sucesso neste ciclo da MainTask.
	7	bApplicationIncompatible	TRUE – A aplicação não é compatível entre os dois CPs. Foi efetuado o download de uma nova aplicação para um dos CP com uma das seguintes alterações: Modificação da área de dados redundantes; Modificação de variáveis simbólicas redundantes. Enquanto este diagnóstico estiver ligado, um dos CPs ficará em estado Inativo até que a mesma aplicação esteja presente nos dois CPs. Isto implica em recarregar a aplicação antiga nos dois CPs ou atualizar os dois CPs com a nova aplicação. Para mais informações sobre como proceder, consultar capítulo Carga de Programas em um CP Redundante.
			FALSE – A aplicação é compatível entre os dois CPs.
%QB(n+5)	0	Reservado	Bit reservado.
	1	bProjectSyncDisable	TRUE – Não será efetuado o sincronismo do projeto aplicativo e do Project archive entre os dois CPs. Trata-se de uma cópia da variável não volátil utilizada para desabilitar a sincronização de projetos, conforme descrito na seção Desabilitação da Sincronização de Projetos. A sincronização de projetos pode ser desabilitada tanto no CP local como no remoto, ou seja, basta executar o comando de desabilitação em um dos CPs para que a sincronização de projetos seja desabilitada. Os comandos de desabilitação do sincronismo de projetos são descritos na seção Desabilitação da Sincronização de Projetos.
			FALSE – Será efetuado o sincronismo do projeto aplicativo e do Project archive entre os dois CPs.
	2	bIncompatibleFirmware	TRUE – A versão de firmware é incompatível entre este CP e o CP remoto.
			FALSE – A versão de firmware é compatível entre este CP ativo e o CP remoto.
	3	bApplicationProjectDiff	TRUE – O projeto aplicativo deste CP é diferente do presente no CP remoto.
			FALSE – O projeto aplicativo deste CP é igual ao do CP remoto.
	4	bProjectArchiveDiff	TRUE – O Project archive deste CP é diferente do presente no CP remoto.
			FALSE – O Project archive deste CP é igual ao do CP remoto.
	5	bOnlineChangeApply	TRUE – Foi realizada alguma alteração online na aplicação e esta ainda não foi sincronizada com o CP reserva.
			FALSE – Não foram realizadas alterações online na aplicação ou estas já foram sincronizadas

			com o CP reserva.
	6	bFailedRED	TRUE – Falha no módulo NX4010. A UCP NX3030 não consegue comunicar-se com este módulo através do barramento, ou existe uma falha no microprocessador do NX4010.
			FALSE – O módulo NX4010 está em correto funcionamento.
	7	bFailedPBUS1A	TRUE – Este CP não consegue comunicar-se no papel de mestre (modo ativo ou passivo) na rede PROFIBUS 1 A. O modo ativo (comunicando com escravos) é assumido pelo CP Ativo. O modo passivo (comunicando com o mestre em modo ativo) é assumido pelo CP Não-Ativo. Esta falha também pode ser indicada caso o módulo NX5001 tenha falha em seu microprocessador, ou caso não consiga se comunicar com a UCP NX3030 via barramento.
			FALSE – Não existem falhas na rede PROFIBUS 1 A.
%QB(n+6)	0	bFailedPBUS1B	TRUE – Este CP não consegue comunicar-se no papel de mestre (modo ativo ou passivo) na rede PROFIBUS 1 B. O modo ativo (comunicando com escravos) é assumido pelo CP Ativo. O modo passivo (comunicando com o mestre em modo ativo) é assumido pelo CP Não-Ativo. Esta falha também pode ser indicada caso o módulo NX5001 tenha falha em seu microprocessador, ou caso não consiga se comunicar com a UCP NX3030 via barramento.
			FALSE – Não existem falhas na rede PROFIBUS 1 B.
	1	bFailureProfibus_1	TRUE – Este CP não consegue comunicar-se no papel de mestre (modo ativo ou passivo) na rede PROFIBUS 1. Caso a rede PROFIBUS 1 seja redundante, FailureProfibus_1 resulta de um "E lógico" entre FailedPBUS1A e FailedPBUS1B. Caso a rede PROFIBUS 1 não seja redundante, FailureProfibus_1 é uma cópia de FailedPBUS1A.
			FALSE – Não existem falhas na rede PROFIBUS 1.
	2	bFailedPBUS2A	TRUE – Este CP não consegue comunicar-se no papel de mestre (modo ativo ou passivo) na rede PROFIBUS 2 A. O modo ativo (comunicando com escravos) é assumido pelo CP Ativo. O modo passivo (comunicando com o mestre em modo ativo) é assumido pelo CP Não-Ativo. Esta falha também pode ser indicada caso o módulo NX5001 tenha falha em seu microprocessador, ou caso não consiga se comunicar com a UCP NX3030 via barramento.
			FALSE – Não existem falhas na rede PROFIBUS 2 A.
	3	bFailedPBUS2B	TRUE – Este CP não consegue comunicar-se no papel de mestre (modo ativo ou passivo) na rede PROFIBUS 2 B. O modo ativo (comunicando com escravos) é assumido pelo CP Ativo. O modo passivo (comunicando com o mestre em modo ativo) é assumido pelo CP Não-Ativo. Esta falha também pode ser indicada caso o módulo NX5001 tenha falha em seu microprocessador, ou caso não consiga se comunicar com a UCP NX3030 via barramento.
			FALSE – Não existem falhas na rede PROFIBUS 2 B.
	4	bFailureProfibus_2	TRUE – Este CP não consegue comunicar-se no papel de mestre (modo ativo ou passivo) na rede PROFIBUS 2. Caso a rede PROFIBUS 2 seja redundante, FailureProfibus_2 resulta de um "E lógico" entre FailedPBUS2A e FailedPBUS2B. Caso a rede PROFIBUS 2 não seja redundante,

			FailureProfibus_2 é uma cópia de FailedPBUS2A.
			FALSE – Não existem falhas na rede PROFIBUS 2.
	5	bProfibusVitalFailureAny	TRUE – Este CP não consegue comunicar-se no papel de mestre (modo ativo ou passivo) com pelo menos uma das redes PROFIBUS configuradas em modo de falha vital.
			FALSE – Não existem falhas nas redes PROFIBUS configuradas em modo de falha vital.
	6	bProfibusVitalFailureAll	TRUE – Este CP não consegue comunicar-se no papel de mestre (modo ativo ou passivo) com todas as redes PROFIBUS configuradas em modo de falha vital.
			FALSE – Não existem falhas nas redes PROFIBUS configuradas em modo de falha vital.
	7	bTurnOffOtherPLC_Normal	TRUE – Este CP está fechando o relé do PX2612 para manter o outro CP desligado em condições normais, e não devido ao modo teste do painel PX2612.
			FALSE – O relé do PX2612 está ligado (bTurnOffOtherPLC_TestMode) ou desligado.
%QB(n+7)	0	bTurnOffOtherPLC_TestMode	TRUE – Este CP está fechando o relé do PX2612 para manter o outro CP desligado devido ao modo teste do painel PX2612.
			FALSE – O relé do PX2612 está ligado (bTurnOffOtherPLC_Normal) ou desligado.
	1	bActiveLED	TRUE – O LED ACTIVE do PX2612 está ligado.
			FALSE – O LED ACTIVE do PX2612 está piscando (bBlinkActiveLED) ou desligado.
	2	bBlinkActiveLED	TRUE – O LED ACTIVE do PX2612 está piscando.
			FALSE – O LED ACTIVE do PX2612 está ligado (bActiveLED) ou desligado.
	3	bStandbyLED	TRUE – O LED STAND-BY do PX2612 está ligado.
			FALSE – O LED STAND-BY do PX2612 está piscando (bBlinkStandbyLED) ou desligado.
	4	bBlinkStandbyLED	TRUE – O LED STAND-BY do PX2612 está piscando.
			FALSE – O LED STAND-BY do PX2612 está ligado (bStandbyLED) ou desligado.
	5	bInactiveLED	TRUE – O LED INACTIVE do PX2612 está ligado.
			FALSE – O LED INACTIVE do PX2612 está desligado ou piscando (bBlinkInactiveLED).
	6	bBlinkInactiveLED	TRUE – O LED INACTIVE do PX2612 está piscando.
			FALSE – O LED INACTIVE do PX2612 está ligado (bInactiveLED) ou desligado.
	7	bRedPanelTestMode	TRUE – O painel PX2612 está em modo teste.
			FALSE – O painel PX2612 está em modo normal.
%QB(n+8)	-	ePLC_ID	Este diagnóstico informa a identificação deste CP: - 0 = não-redundante - 2 = CPA - 3 = CPB Trata-se de uma cópia da variável não-volátil utilizada para identificar o CP, conforme descrito na seção Identificação de uma UCP NX3030. Na seção Carga Inicial de um Projeto Redundante descreve-se o comando do MasterTool utilizado para escrever sobre esta variável não-volátil.
%QB(n+9)	-	eRedState	Informa o estado de redundância deste CP:

			- Não-Configurado = 0 - Inicializando = 2 - Reserva = 3 - Ativo = 4 - Inativo = 5
%QB(n+10)	-	ePreviousRedState	Valor que o diagnóstico RedState possuía antes da última transição de estados.
%QW(n+11)	-	wRedStateDuration	Mede há quanto tempo (milissegundos) o estado de redundância atual foi assumido. Este tempo para de incrementar quanto atinge 65535 ms.
%QW(n+13)	-	wConfigErrorCode	Código de erro descoberto durante o processo de configuração no estado Não-Configurado. Ver diagnóstico ConfigError descrito anteriormente.
%QD(n+15)	-	dwApplicationCRC	CRC de 32 bits da aplicação, utilizado para detectar diferenças entre as aplicações dos 2 CPs.
%QD(n+19)	-	dwArchiveCRC	CRC de 32 bits do project archive, utilizado para detectar diferenças entre os project archives dos 2 CPs.
%QD(n+23)	-	dwFirmwareVersion	Versão de firmware deste CP, utilizada para verificar compatibilidade entre firmware dos 2 CPs.
%QD(n+27)	-	dwIECTimer	A sincronização do IEC Timer é necessária para operação bump-less de alguns blocos de função como TON e TOF. Através deste diagnóstico o IEC Timer do CP Ativo é recebido e atualizado no CP Não-Ativo, desde que o serviço Troca de Diagnósticos e Comandos tenha sido executado com sucesso. Sua contagem inicia em 0 e incrementa até 4294967295. Após estouro de contagem, reinicia com valor 0.
%QW(n+31)	-	wCycleCounter	Contador de 16 bits utilizado como informação auxiliar de sequência nos Logs de Eventos da Redundância. No CP Ativo, é incrementado a cada ciclo da MainTask. No CP Não-Ativo, recebe uma cópia do valor existente no CP Ativo, desde que o serviço Troca de Diagnósticos e Comandos tenha sido executado com sucesso. Sua contagem inicia em 0 e incrementa até 65535. Após estouro de contagem, reinicia com valor 0.

Tabela 6-6. Diagnósticos Gerais da Redundância

Notas:

Visualização das Estruturas de Diagnóstico: As Estruturas de Diagnóstico adicionadas ao projeto podem ser visualizadas acessando o item "Library Manager" na treeview da janela do MasterTool IEC XE. Com isso, é possível visualizar todos os datatypes definidos na estrutura.

Variável de representação direta: O "n" representa o valor configurado no módulo NX4010, através do software MasterTool IEC XE, como Endereço Inicial de Diagnósticos em %Q. Esta definição vale para todas as estruturas de diagnóstico.

Diretiva AT: A diretiva AT é uma palavra reservada no software programador, que se relaciona a um endereço de uma variável. No caso do módulo NX4010 a variável DG_NX4010 está relacionada ao endereço inicial de diagnósticos do módulo.

A sub-estrutura "sNETA_Diag" possui os seguintes campos para diagnósticos dos canais de sincronismo NETA:

DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.bExchangeSync: Quando esta variável de diagnóstico estiver com o valor igual a FALSE não é possível definir o estado de alguns outros diagnósticos, como por exemplo bIncompatibleFirmware, bApplicationProjectDiff e

bProjectArchiveDiff. Não irão representar o valor correto pois dependem do correto funcionamento da comunicação entre os dois CPs para que a informação possa ser corretamente gerada.

Variável de Representação Direta		Variável AT DG_NX4010.tRedundancy.RedDgnLoc.sNETA_Diag.*	Descrição
Variável	**Bit**		
%QB(n+33)	0	bGeneralFailure	TRUE – O canal de sincronismo possui algum tipo de falha. Os 3 próximos diagnósticos indicarão a falha específica.
			FALSE – O canal de sincronismo está em correto funcionamento.
	1	bInternalFailure	TRUE – A falha detectada tem sua causa localizada dentro deste CP. Tais falhas são tratadas de forma diferenciada.
			FALSE – O módulo NX4010 está em correto funcionamento.
	2	bLinkDownFailure	TRUE – O cabo AL-2319A está desconectado do módulo NX4010 ou rompido, em um dos 2 CPs.
			FALSE – O cabo AL-2319A está conectado ao módulo NX4010.
	3	bTimeoutFailure	TRUE – Esta falha é reportada caso um serviço de sincronização não tenha terminado com sucesso até um time-out especificado, e não tenham sido encontradas falhas do tipo bInternalFailure ou bLinkDownFailure que justificassem isso.
			FALSE – O módulo NX4010 está em correto funcionamento.
	4 a 7	bReserved[4..7]	Reservados.

Tabela 6-7. Diagnósticos Específicos da Interface NETA

A sub-estrutura “sNETB_Diag” possui os seguintes campos para diagnósticos dos canais de sincronismo NETB:

Variável de Representação Direta		Variável AT DG_NX4010.tRedundancy.RedDgnLoc.sNETB_Diag.*	Descrição
Variável	**Bit**		
%QB(n+34)	0	bGeneralFailure	TRUE – O canal de sincronismo possui algum tipo de falha. Os 3 próximos diagnósticos indicarão a falha específica.
			FALSE – O canal de sincronismo está em correto funcionamento.
	1	bInternalFailure	TRUE – A falha detectada tem sua causa localizada dentro deste CP. Tais falhas são tratadas de forma diferenciada.
			FALSE – O módulo NX4010 está em correto funcionamento.
	2	bLinkDownFailure	TRUE – O cabo AL-2319B está desconectado do módulo NX4010 ou rompido, em um dos 2 CPs.
			FALSE – O cabo AL-2319 está conectado ao módulo NX4010.
	3	bTimeoutFailure	TRUE – Esta falha é reportada caso um serviço de sincronização não tenha terminado com sucesso até um time-out especificado, e não tenham sido encontradas falhas do tipo bInternalFailure ou bLinkDownFailure que justificassem isso.
			FALSE – O módulo NX4010 está em correto funcionamento.

	4 a 7	bReserved[4..7]	Reservados.

Tabela 6-8. Diagnósticos Específicos da Interface NETB

A sub-estrutura "sNET_Stat" contém estatísticas de falhas e sucessos dos serviços. As estatísticas dos CPs local e remoto podem ser reiniciadas através dos comandos:

```
//CP Local
DG_NX4010.tRedundancy.RedCmdLoc.bResetNETStatisticsLocal := TRUE;
//CP Remoto
DG_NX4010.tRedundancy.RedCmdLoc.bResetNETStatisticsRemote := TRUE;
```

A sub-estrutura contém os seguintes contadores:

Variável de Representação Direta	Variável AT DG_NX4010.tRedundancy. RedDgnLoc.sNET_Stat.*	Descrição
%QW(n+35)	wSuccessExchDgCmdSync	Contagem de sucessos do serviço Troca de Diagnósticos e Comandos. (0 a 65535)
%QW(n+37)	wFailedExchDgCmdSync	Contagem de falhas do serviço Troca de Diagnósticos e Comandos. (0 a 65535)
%QW(n+39)	wSuccessRedDataSync	Contagem de sucessos do serviço Sincronização de Dados Redundantes. (0 a 65535)
%QW(n+41)	wFailedRedDataSync	Contagem de falhas do serviço Sincronização de Dados Redundantes. (0 a 65535)
%QW(n+43)	wSuccessRedForceSync	Contagem de sucessos do serviço Sincronização de Lista de Forçamentos Redundantes. (0 a 65535)
%QW(n+45)	wFailedRedForceSync	Contagem de falhas do serviço Sincronização de Lista de Forçamentos Redundantes. (0 a 65535)
%QB(n+47)	byReserved[1..8]	Reservados

Tabela 6-9. Diagnósticos Específicos da Interface

Nota:

Contadores: Todos os contadores retornam a zero quando o seu valor limite é ultrapassado.

Comandos da Redundância

Os campos de comandos das estruturas RedCmdLoc e RedCmdRem possuem sempre um sufixo que pode ser Local ou Remote. Por exemplo, existem os campos de comando StandbyLocal e StandbyRemote, que têm efeito equivalente ao botão STAND-BY do painel PX2612.

Um comando com sufixo Local gerado em RedCmdLoc será executado no próprio CP (local).

Por outro lado, um comando com sufixo Remote gerado em RedCmdLoc será executado no outro CP (remoto). Isto funciona da seguinte maneira:

- O CP remoto, a cada ciclo da MainTask, recebe uma cópia de RedCmdLoc do CP local via NETA / NETB, e esta cópia é chamada de RedCmdRem no CP remoto
- O CP remoto somente executa comandos de RedCmdRem que tenham o sufixo Remote

Exemplo 1: se o CP local estiver em estado Ativo, e deseja-se chaveá-lo para o estado Reserva, pode-se ligar o bit DG_NX4010.tRedundancy.RedCmdLoc.bStandbyLocal no CP local.

Exemplo 2: se o CP remoto estiver em estado Ativo, e deseja-se chaveá-lo para o estado Reserva, pode-se ligar o bit DG_NX4010.tRedundancy.RedCmdLoc.bStandbyRemote no CP local. Isto pode ser útil, por exemplo, se a comunicação de um sistema SCADA não está disponível temporariamente

com o CP remoto. Neste caso, o comando é escrito pelo SCADA no CP local, que o repassa para o CP remoto via NETA / NETB.

ATENÇÃO:
Se diagnóstico DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.bExchangeSync estiver indicando falha no serviço Troca de Diagnósticos e Comandos, um comando com sufixo Remote não poderá ser repassado para o CP remoto, e portanto não será executado.

Para disparar um comando, deve-se sempre ligar o bit correspondente em RedCmdLoc. Isto pode ser feito por um sistema SCADA, fazendo uma escrita via MasterTool, ou até mesmo ligando o bit dentro de uma POU como ActivePrg ou NonSkippedPrg.

O usuário não precisa se preocupar com o desligamento do bit de comando, que será feito automaticamente pelo gerenciador de redundância:

- No caso de comandos executados no CP local (RedCmdLoc + comando com sufixo Local), o bit é desligado assim que o comando for percebido e executado
- No caso de comandos executados no CP remoto (RedCmdRem + comando com sufixo Remote):
  - no CP remoto, o comando é executado quando o gerenciador de redundância percebe uma borda de subida no bit de comando
  - no CP local onde o comando foi gerado, o bit é desligado automaticamente no próximo ciclo da MainTask

ATENÇÃO:
Existem dois bits de comando que normalmente devem ser desligados pelo usuário: DG_NX4010.tRedundancy.RedCmdLoc.bTestModeLocal e DG_NX4010.tRedundancy. RedCmdLoc.bTestRelayLocal. Ver maiores detalhes sobre estes comandos adiante nesta seção. Caso o usuário esqueça de desligá-los, existem mecanismos automáticos previstos para desligá-los.

É importante salientar que quaisquer ligamentos ou desligamentos de comandos serão registrados nos logs de eventos da redundância, o que é importante para avaliação histórica, por exemplo, para determinar as causas de um switchover.

A seguir, são definidos os campos das estruturas RedCmdLoc e RedCmdRem:

Variável de Representação Direta		**Variável AT DG_NX4010.tRedundancy.RedCmdLoc.***	**Descrição**
Variável	**Bit**		
%QB(n+55)	0	bButtonTurnOnLocal	TRUE – Trata-se de uma cópia processada do botão TURN ON PLCx lido do painel PX2612. Este bit só é ligado um segundo depois do pressionamento do botão, e desligado imediatamente ao liberar o botão. É importante ressaltar que esse bit será ligado quando o botão TURN ON do CP remoto estiver pressionado, visto que esse tipo de comando sempre é enviado do CP remoto.
			FALSE – O botão TURN ON PLCx não está pressionado.
	1	bButtonStandbyLocal	TRUE – Trata-se de uma cópia processada do botão STAND-BY lido do painel PX2612. Este bit só é ligado um segundo depois do pressionamento do botão, e desligado imediatamente ao liberar o botão.
			FALSE – O botão STAND-BY não está pressionado.
	2	bButtonInactiveLocal	TRUE – Trata-se de uma cópia processada do botão INACTIVE lido do painel PX2612. Este bit só é ligado um segundo depois do pressionamento do botão, e desligado imediatamente ao liberar o botão.

			FALSE – O botão INACTIVE não está pressionado.
	3	bAutoConfigLocal	TRUE – Este diagnóstico informa uma solicitação de configuração automática, necessária para deixar o estado Não-Configurado em algumas situações.
			FALSE – Solicitação de configuração automática desabilitada.
	4	bTurnOnLocal	TRUE – Este comando produz uma ação equivalente ao botão TURN ON PLCX do PX2612, no CP local.
			FALSE – O botão TURN ON PLCx do CP local não está pressionado.
	5	bStandbyLocal	TRUE – Este comando produz uma ação equivalente ao botão STAND-BY do PX2612, no CP local.
			FALSE – O botão STAND-BY do CP local não está pressionado.
	6	bInactiveLocal	TRUE – Este comando produz uma ação equivalente ao botão INACTIVE do PX2612, no CP local.
			FALSE – O botão INACTIVE do CP local não está pressionado.
	7	bResetNETStatisticsLocal	TRUE – Este comando reseta as estatísticas de NETA / NETB (veja sub-estrutura SNET_Stat em RedDgnLoc e RedDgnRem). Tais estatísticas são contadores de falhas e sucessos em serviços de sincronização.
			FALSE – O comando de reset das estatísticas de NETA / NETB no CP local não foi acionado.
%QB(n+56)	0	bTestModeLocal	TRUE – Este comando coloca o painel PX2612 em modo teste, permitindo que seus componentes sejam testados (LEDs, relés e botões), conforme explica-se na seção Teste do Painel PX2612. O modo teste do PX2612 só é efetivamente aceito quando este bit é ligado nos dois CPs. Ou seja, para que o PX2612 esteja em modo teste, o CP verifica se RedCmdLoc.bTestModeLocal e se RedCmdRem.bTestModeLocal estão ambos ligados. O diagnóstico RedDgnLoc.bRedPanelTestMode é ligado para informar que o PX2612 está realmente em modo teste. Normalmente, o usuário deve desligar o bit bTestModeLocal nos 2 CPs assim que concluir os testes do PX2612, mas caso esqueça de fazer isto, o bit será desligado automaticamente 15 minutos depois de ser ligado.
			FALSE – O comando que coloca o painel PX2612 em modo teste está desativado.
	1	bTestRelayLocal	TRUE – Este comando é usado para testar o relé NA do PX2612, e consequentemente também o relé NF externo, usados para um eventual desligamento do outro CP. Este comando só é aceito enquanto o PX2612 está em modo teste, sendo automaticamente desligado e ignorado caso o PX2612 não esteja em modo teste. Normalmente, o usuário deve desligar o bit bTestRelayLocal assim que concluir o teste do relé, mas caso esqueça de fazer isto, o bit será desligado assim que o modo teste for encerrado. É importante também salientar que este comando só é aceito no CP Ativo, para evitar que o CP Não-Ativo desligue o CP Ativo.
			FALSE – O comando utilizado para testar o relé NA do PX2612 está desativado.
	2	bStandbyRemote	TRUE – Este comando produz uma ação

			equivalente ao botão STAND-BY do PX2612, no CP remoto.
			FALSE – O botão STAND-BY do CP remoto não está pressionado.
	3	bInactiveRemote	TRUE – Este comando produz uma ação equivalente ao botão INACTIVE do PX2612, no CP remoto.
			FALSE – O botão INACTIVE do CP local não está pressionado.
	4	bResetNETStatisticsRemote	TRUE – Produz uma ação equivalente ao comando ResetNETStatisticsLocal, porém desta vez no CP remoto.
			FALSE – O comando de reset das estatísticas de NETA / NETB no CP remoto não foi acionado.
	5 a 7	bReserved[5..7]	Reservados.

Tabela 6-10. Comandos da Redundância

Informações do Usuário Trocados entre CPA e CPB

O serviço de sincronização Troca de Diagnósticos e Comando, troca as seguintes estruturas de dados entre os dois CPs em cada ciclo da MainTask, usando os canais de sincronismo NETA / NETB:

- Diagnósticos de Redundância (RedDgnLoc e RedDgnRem), já discutidos na seção Estrutura de Diagnósticos da Redundância
- Comandos de Redundância (RedCmdLoc e RedCmdRem), já discutidos na seção Comandos da Redundância
- Informações do Usuário Trocados entre CPA e CPB (RedUsrLoc e RedUsrRem), que serão discutidos nesta seção

As estruturas RedUsrLoc e RedUsrRem são simplesmente um array de 128 bytes, cuja utilização pode ser livremente definida pelo usuário. Elas permitem que o usuário transfira, a cada ciclo, 128 bytes de informação do CPA para o CPB, e outros 128 bytes do CPB para o CPA.

RedUsrRem é uma cópia de RedUsrLoc do outro CP, recebida via NETA / NETB. Determinado CP escreve informações em RedUsrLoc, que serão lidas no outro CP em RedUsrRem.

Estas estruturas de dados podem ter diversas utilidades. Por exemplo, supondo que o sistema SCADA se conecte somente ao CP Ativo, e que deseja-se visualizar algumas informações do CP Não-Ativo, o CP Não-Ativo pode colocar estas informações nesta estrutura de dados. Entre tais informações, por exemplo, podem constar alguns diagnósticos que não estejam mapeados em RedDgnLoc e RedDgnRem.

Diagnósticos MODBUS utilizados na Redundância

Para verificar se existe falha em todos os MODBUS Server configurados em uma instância MODBUS Client, existe um diagnóstico especifico em cada instância MODBUS Client configurada. Na Tabela 6-11, é exibido o diagnóstico para este tipo de falha em uma instância MODBUS Client chamada MODBUS_Symbol_Client.

Variável DG_MODBUS_Symbol_Client.tDiag.*	Descrição
bAllDevicesCommFailure	TRUE – Todos os Servers configurados neste Client estão apresentando erro.
	FALSE – Existe pelo menos um Server configurado neste Client que não está apresentando erro.

Tabela 6-11. Diagnóstico MODBUS Client

Quando configurado com falha vital Ethernet, este diagnóstico é consultado e 3 segundos após a alteração do seu estado de FALSE para TRUE, ocorre um switchover caso as outras condições para este evento estejam satisfeitas.

Logs de Evento da Redundância

O MasterTool permite observar diversos logs para um CP Nexto, entre os quais encontra-se os Logs de Eventos da Redundância. Estas mensagens, específicas para a redundância, registram no Log de Sistema modificações em praticamente todos os campos das estruturas de dados de diagnósticos e comandos de redundância, que são as seguintes:

- RedDgnLoc
- RedDgnRem
- RedCmdLoc
- RedCmdRem

No caso das estruturas de diagnósticos, apenas os seguintes campos não geram logs de eventos:

- wRedStateDuration
- wCycleCounter
- dwIECTimer
- SyncLinkStatistics NET_Stat

Cada linha mostrada no log possui as seguintes colunas:

- Marca de Tempo: data e hora do evento, com resolução de milissegundos
- Severidade: informação, advertência, erro ou exceção
- Descrição: texto que descreve o evento
- Componente: componente que gerou o evento, que no caso do Redundancy Event Log, será o "Redundancy Management"

O texto na coluna "Descrição" contém as seguintes informações:

- O nome do diagnóstico ou comando que sofreu modificação
- O novo valor assumido por este diagnóstico ou comando
- O valor de RedDgnLoclsGeneral_Diag.wCycleCounter, que pode ser usado como informação auxiliar de sequência
- O valor de RedDgnRem.sGeneral_Diag.CycleCounter, que pode ser usado como informação auxiliar de sequência

Um exemplo da coluna Descrição poderia ser o seguinte:

```
RedDgnLoc.sGeneral_Diag.eRedStat = Active [Local cycle = 1234, Remote
cycle = 1233].
```

Para acessar esta tela, deve-se dar um duplo clique sobre o dispositivo (NX3030) na árvore de dispositivos, e depois selecionar a aba "Log". Existe um filtro que permite selecionar somente o componente "Redundancy Management", para exibir somente os eventos de redundância.

ATENÇÃO: Alguns diagnósticos podem apontar possíveis falhas durante a inicialização do sistema redundante e nos primeiro ciclos das tarefas. Mas, em um correto funcionamento do sistema, estes diagnósticos voltam a indicar a ausência de erros logo após a inicialização do sistema.

Teste do Painel PX2612

O painel PX2612 é constituído de botões, LEDs e relés. Muitos destes recursos são pouco utilizados, e portanto raramente são testados para perceber que possuem algum defeito. É importante que, periodicamente, se façam testes para verificar se estes recursos estão funcionando, para evitar que falhas permaneçam ocultas e impeçam o uso do PX2612 quando o mesmo for necessário.

Entrada no Modo Teste

O primeiro passo para testar o PX2612 é colocá-lo em modo teste. Isto é feito ligando o bit de comando DG_NX4010.tRedundancy.RedCmdLoc.bTestModeLocal nos dois CPs.

Determinado CP perceberá que está em modo teste sempre que os dois seguintes bits estiverem ligados:

- DG_NX4010.tRedundancy.RedCmdLoc.bTestModeLocal (RedCmdLoc.bTestModeLocal ligado neste CP)
- DG_NX4010.tRedundancy.RedCmdRem.bTestModeLocal (RedCmdLoc.bTestModeLocal ligado no outro CP)

Quando estes dois bits estão ligados, o CP liga o bit de diagnóstico DG_NX4010.tRedundancy.RedDgnLoc.sGeneral_Diag.bRedPanelTestMode, para informar que o PX2612 está em modo teste.

Saídas Manual e Automática do Modo Teste

O usuário pode encerrar o modo teste manualmente, desligando o bit DG_NX4010.tRedundancy.RedCmdLoc.bTestModeLocal nos dois CPs. Na verdade, basta desligá-lo em um dos CPs, pois o modo teste exige que este bit esteja ligado nos dois CPs. No entanto, aconselha-se desligá-lo nos dois CPs.

Caso o usuário esqueça de desligar DG_NX4010.tRedundancy.RedCmdLoc.bTestModeLocal, este bit é desligado automaticamente, 15 minutos depois de ser ligado, encerrando automaticamente o modo teste.

Teste dos LEDs

Durante o modo teste, os seis LEDs devem piscar, perdendo sua utilidade normal, que é mostrar o estado da redundância.

Teste dos Botões

Ao pressionar um botão no modo teste, um LED correspondente irá parar de piscar, e ficará aceso. A Tabela 6-12 mostra a correspondência entre o botão pressionado e o LED que permanecerá aceso.

Botão Testado	LED Correspondente
TURN ON PLC A	ACTIVE – PLC B
STAND-BY PLC A	STAND-BY PLC A
INACTIVE PLC A	INACTIVE PLC A
TURN ON PLC B	ACTIVE – PLC A
STAND-BY PLC B	STAND-BY PLC B
INACTIVE PLC B	INACTIVE PLC B

Tabela 6-12. Correspondência entre Botões e LEDs no Teste de Botões do PX2612

Observa-se que, normalmente, o LED fica ao lado do botão pressionado, exceto para os botões TURN ON PLCx.

Antes de o LED permanecer aceso, é necessário manter o botão pressionado por, no mínimo, um segundo. O LED volta a piscar assim que o botão for liberado.

Durante o modo teste, os botões não permitem executar as funções que executariam fora do modo teste, tais como provocar uma mudança de estado de redundância.

Teste dos Relés

Para testar os relés, foi criado o bit de comando DG_NX4010.tRedundancy.RedCmdLoc.bTestRelayLocal. Ao ligar este bit, se este CP estiver em modo teste, e além disso estiver em estado Ativo, ele acionará o relé, o que deverá provocar o

desligamento do outro CP (Não-Ativo). Desligando DG_NX4010.tRedundancy.RedCmdLoc.bTestRelayLocal, o relé é liberado, permitindo o religamento do outro CP.

O comando não tem efeito no CP Não-Ativo, para evitar que o mesmo desligue o CP Ativo.

O usuário deverá provocar um switchover entre CPs (Ativo X Não-Ativo), para testar o relé nos dois CPs.

Quando o CP que foi desligado é religado e reinicializado, ele volta com DG_NX4010.tRedundancy.RedCmdLoc.bTestModeLocal desligado, portanto, o modo teste está cancelado. Deve-se ligar novamente o bit DG_NX4010.tRedundancy.RedCmdLoc.bTestModeLocal neste CP, e passá-lo para o estado Ativo, antes de testar o seu relé.

Quando o modo teste é encerrado, o bit de comando DG_NX4010.tRedundancy.RedCmdLoc.bTestRelayLocal é automaticamente desligado, caso o usuário o tenha esquecido ligado.

Sequência Sugerida para Executar os Testes do PX2612

A seguinte sequência é sugerida para executar os testes do PX2612:

- Ligar o bit de comando DG_NX4010.tRedundancy.RedCmdLoc.bTestModeLocal nos dois CPs (CPA e CPB).
- Deve-se observar que os 6 LEDs estão piscando.
- Pressionar, um a um, os 6 botões, e verificar se o LED correspondente para de piscar e permanece aceso enquanto o botão está pressionado. Deve-se lembrar que o botão deve ficar pressionado no mínimo por um segundo antes que o LED pare de piscar e fique aceso, e que o LED voltará a piscar assim que o botão for liberado.
- Ligar o bit de comando DG_NX4010.tRedundancy.RedCmdLoc.bTestRelayLocal no CP Ativo. Deve-se observar o desligamento do CP Não-Ativo.
- Desligar o bit de comando DG_NX4010.tRedundancy.RedCmdLoc.bTestRelayLocal no CP Ativo. Deve-se observar o religamento do CP Não-Ativo.
- Aguardar que o CP Não-Ativo seja reinicializado e assuma o estado Reserva. O modo teste não estará ativo, pois o bit DG_NX4010.tRedundancy.RedCmdLoc.bTestModeLocal foi desligado na inicialização do CP que agora está em Reserva.
- Provocar um switchover entre CPs, pressionando o botão STAND-BY do CP Ativo. O uso normal do botão STAND-BY é possível porque o modo teste não está ativo.
- Ligar o bit de comando DG_NX4010.tRedundancy.RedCmdLoc.bTestModeLocal no novo CP Ativo, que acabou de sair do modo Reserva. Desta forma, o modo teste volta a estar ativo. Caso o bit do CP Reserva esteja desligado, o mesmo deve ser ligado novamente.
- Ligar o bit de comando DG_NX4010.tRedundancy.RedCmdLoc.bTestRelayLocal no CP Ativo. Deve-se observar o desligamento do CP Não-Ativo.
- Desligar o bit de comando DG_NX4010.tRedundancy.RedCmdLoc.bTestRelayLocal no CP Ativo. Deve-se observar o religamento do CP Não-Ativo.
- Desligar o bit de comando DG_NX4010.tRedundancy.RedCmdLoc.bTestModeLocal no CP Ativo, para encerrar o modo teste. Não é necessário fazer isto no CP Reserva, pois o mesmo acabou de ser inicializado, com o bit DG_NX4010.tRedundancy.RedCmdLoc.bTestModeLocal desligado.

7. Manutenção

Diagnósticos do Módulo

Uma das características da Série Nexto é a geração de diagnósticos de comportamentos atípicos, sejam eles falhas, erros ou modos de operação, possibilitando ao operador identificar e solucionar problemas que venham a ocorrer com o sistema com grande facilidade.

As UCPs Nexto proporcionam diversas maneiras de se visualizar os diagnósticos gerados pelo sistema, são elas:

- One Touch Diag
- Diagnósticos via LED
- Diagnósticos via WEB
- Diagnósticos via Variáveis
- Diagnósticos via Blocos Funcionais

O primeiro deles é uma característica inovadora da Série Nexto, a qual possibilita um rápido acesso às condições anormais da aplicação. O segundo é puramente visual, gerado através de dois LEDs presentes no painel (DG e WD) e também dos LEDs presentes no conector RJ45 (exclusivos para a conexão Ethernet). A próxima característica é a visualização gráfica em uma página WEB do bastidor e os respectivos módulos configurados, sendo permitido o acesso individual do estado de operação e os diagnósticos ativos. Os diagnósticos também são armazenados diretamente em variáveis da UCP, tanto de representação direta (%Q), como as simbólicas mapeadas através da diretiva AT, e podem ser utilizados pela aplicação do usuário, por exemplo, sendo apresentados em um sistema de supervisão. Os últimos apresentam condições específicas de funcionamento do sistema.

A função destes diagnósticos é apontar possíveis problemas de instalação ou configuração do sistema, e de problemas ou deficiências das redes de comunicação. O capítulo de Manutenção deve ser consultado pelo usuário sempre que necessário.

One Touch Diag

One Touch Diag (OTD), ou seja, diagnóstico com um único pressionamento, é uma característica exclusiva que a Série Nexto traz para os controladores programáveis. Com este novo conceito, o usuário pode verificar os diagnósticos de qualquer módulo presente no sistema diretamente no visor gráfico da UCP, com um único pressionamento no botão de diagnóstico do respectivo módulo. Essa é uma poderosa ferramenta de diagnóstico que pode ser usada off-line (sem sistema de supervisão ou software programador), tornando mais fácil encontrar e resolver rapidamente possíveis problemas.

O botão de diagnóstico está localizado na parte superior da UCP, em local de fácil acesso, e, além de fornecer os diagnósticos ativos, permite o acesso ao menu de navegação, descrito no capítulo Configuração –Menu Informativo e de Configuração da UCP.

A Figura 7-1 mostra a localização da tecla na UCP:

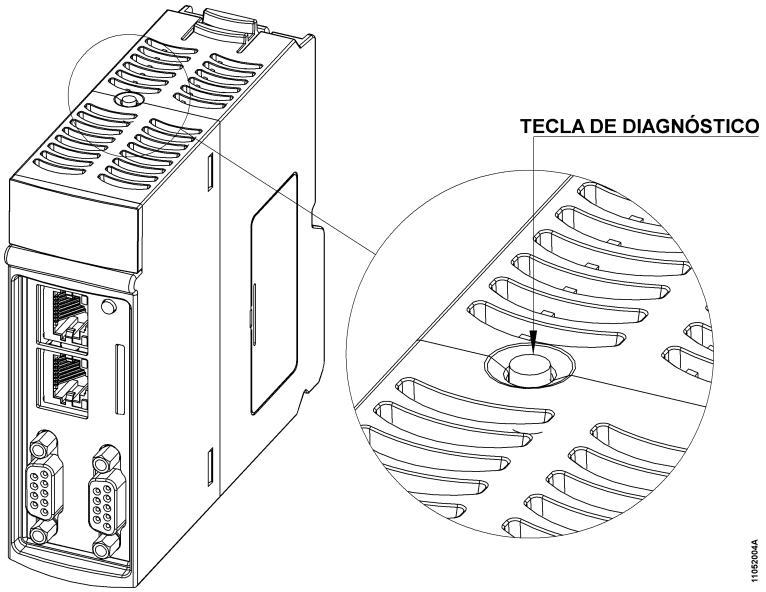

Figura 7-1. Botão de Diagnóstico

Com apenas um pressionamento curto, a UCP começa a mostrar os diagnósticos do barramento (quando ativos, caso contrário exibe a mensagem “SEM DIAG”). Inicialmente, será visualizada a Tag (configurada nas propriedades do módulo no software MasterTool IEC XE, seguindo a IEC 61131-3), ou seja, o nome atribuído à UCP, em seguida, serão mostrados todos os diagnósticos, através de mensagens no visor da UCP. Esse processo será executado por duas vezes no visor. Tudo ocorre de forma automática, sendo que o usuário somente deverá executar o pressionamento curto inicial e a UCP será responsável por exibir os diagnósticos. Os diagnósticos de outros módulos presentes no barramento também serão exibidos no visor gráfico da UCP, através de um pressionamento curto no botão de diagnóstico dos mesmos, no mesmo modelo da apresentação dos diagnósticos da UCP.

A Figura 7-2 mostra todo o processo a partir do pressionamento curto, sendo a condição e os tempos da UCP representados nos retângulos menores. É importante salientar que os diagnósticos poderão ter mais de uma tela, ou seja, o tempo especificado no fluxograma abaixo é válido para cada uma delas.

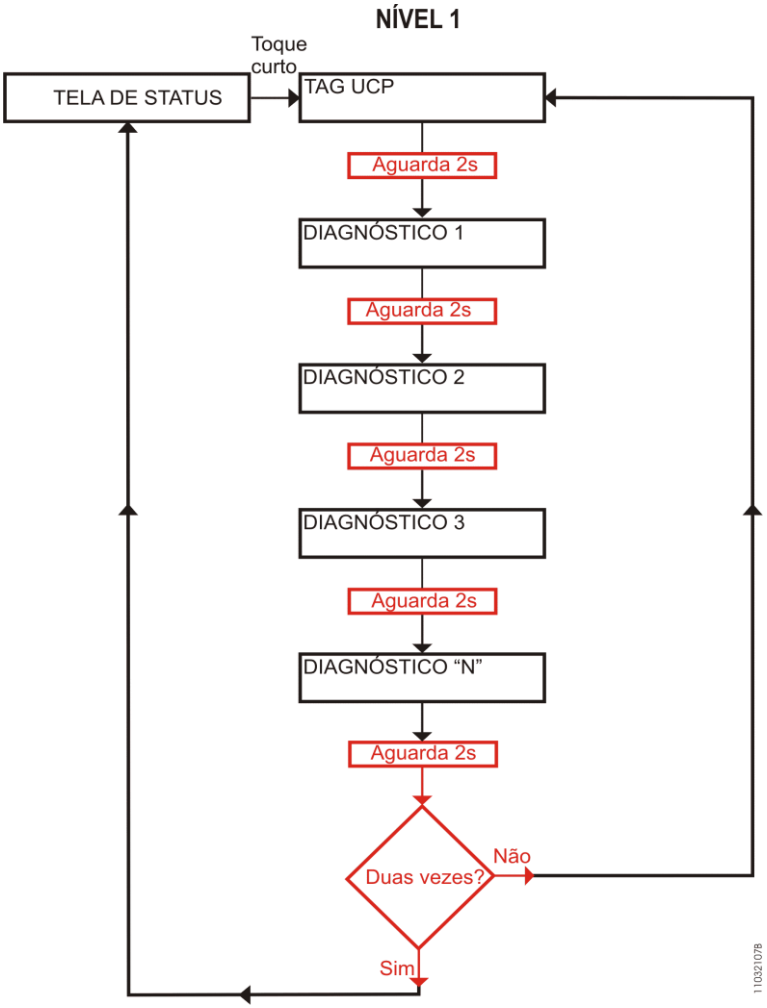

Figura 7-2. Visualização dos Diagnósticos da UCP

Para finalizar, antes de todo o processo de visualização ser efetuado, basta dar um pressionamento curto no botão de diagnóstico, em qualquer momento, ou pressionar o botão diagnóstico de algum módulo de E/S presente no barramento. Também, é importante notar que o One Touch Diag só estará disponível quando o módulo estiver em modo operacional.

Caso seja efetuado um pressionamento longo, a UCP entrará no menu de navegação, o qual está descrito no capítulo Configuração – Menu Informativo e de Configuração da UCP.

A Tabela 7-1 mostra a diferença entre os tempos do pressionamento curto, pressionamento longo e botão preso.

Tipo de Pressionamento	Tempo Mínimo	Tempo Máximo	Condição para Indicação
Sem pressionamento	-	59,99 ms	-
Pressionamento curto	60 ms	0,99 s	Liberação
Pressionamento longo	1 s	20 s	Mais do que 1 s até 20 s
Tecla Presa	20,01 s	(∞)	Indicação em diagnóstico, consultar a Tabela 7-6

Tabela 7-1. Tempo de One Touch

As mensagens exibidas no visor gráfico das UCPs Nexto, correspondentes aos diagnósticos, estão descritas na seção Diagnósticos via Variáveis, na Tabela 7-6.

Caso ocorra uma situação de tecla presa de algum módulo de E/S presente no barramento, o botão de diagnóstico deste deixará de indicar diagnósticos no visor gráfico da UCP quando for pressionado, neste caso a UCP irá indicar que há módulos com diagnóstico ativo. Para que seja possível eliminar este diagnóstico da UCP, será necessário realizar uma troca a quente do módulo com o diagnóstico ativo.

Para mais detalhes sobre o procedimento de visualização dos diagnósticos da UCP ou de outros módulos do barramento, ver descrição no Manual de Utilização Série Nexto – MU214000.

Diagnósticos via LED

As UCPs da Série Nexto possuem um LED para indicação de diagnóstico (LED DG) e um LED para indicar ocorrência de cão-de-guarda (LED WD). A Tabela 7-2 e Tabela 7-3 mostram o significado de cada estado e suas respectivas descrições.

DG (Diagnóstico)

Verde	Vermelho	Descrição	Causas	Prioridade
Desligado	**Desligado**	Não utilizado	Sem fonte de alimentação Problema de Hardware	-
Ligado	**Desligado**	Aplicação em Execução (Modo Run)	-	3 (Baixa)
Desligado	**Ligado**	Aplicação Parada (Modo Stop)	-	3 (Baixa)
Piscando 2x	**Desligado**	Módulos do barramento com diagnóstico	No mínimo um módulo do barramento, incluindo a UCP, está com algum diagnóstico ativo	1
Piscando 3x	**Desligado**	Forçamento de dados	Alguma área de memória está sendo forçada pelo usuário via MasterTool IEC XE	2
Desligado	**Piscando 4x**	Erro de configuração ou de hardware no barramento.	O barramento está danificado ou não está configurado corretamente.	0 (Alta)

Tabela 7-2. Descrição dos Estados do LED de Diagnóstico

WD (Cão-de-guarda)

LED Vermelho	Descrição	Causas	Prioridade
Desligado	Sem indicação de cão-de-guarda	Operação normal	3 (Baixa)
Piscando 1x	Cão-de-guarda de Software	Cão-de-guarda da aplicação de usuário	2
Ligado	Cão-de-guarda de Hardware	Módulo danificado e/ou sistema operacional corrompido	1 (Alta)

Tabela 7-3. Descrição dos Estados do LED de Cão-de-guarda

Notas:

Cão-de-guarda de software: Para remover a indicação de cão-de-guarda, deve-se efetuar um reset da aplicação ou desligar e ligar novamente a UCP. Esse cão-de-guarda ocorre quando o tempo de execução da aplicação de usuário for maior que o tempo de cão-de-guarda configurado.

O diagnóstico pode ser verificado no operando Exception.wExceptionCode, ver Tabela 7-10.

Cão-de-guarda de hardware: Para limpar qualquer indicação de cão-de-guarda, como no LED WD ou no operando Reset.bWatchdogReset, o modulo deve ser desconectado da fonte de alimentação.

Para verificar as condições da aplicação na reinicialização do módulo, ver configurações na Tabela 4-1.

LEDs Conector RJ45

Os dois LEDs presentes nos conectores RJ45 (no caso da NX3010 um conector somente), identificados por NET 1 e NET 2, auxiliam o usuário na detecção de problemas na rede física instalada, indicando a velocidade do link de rede e a existência de tráfego de comunicação com a interface. O significado dos LEDs é apresentado na Tabela 7-4.

Amarelo	Verde	Significado
○	○	Ausência de LINK de rede
●	○	LINK de rede de 10 Mbits/s
●	●	LINK de rede de 100 Mbits/s
X	–	Ocorrência de transmissão ou recepção na rede Ethernet, pelo ou para este endereço IP. Pisca sob demanda da UCP Nexto, e não a cada transmissão ou recepção, ou seja, pode piscar com uma frequência menor que a frequência real de transmissão ou recepção.

Tabela 7-4. Significado dos LEDs Ethernet

Diagnósticos via WEB

Além das características apresentadas anteriormente, a Série Nexto traz para o usuário uma ferramenta inovadora de acesso aos diagnósticos e estados de operação do sistema, através de uma página WEB.

A utilização, além de dinâmica, é bastante intuitiva e facilita as operações do usuário. Em outras palavras, pode substituir o uso de um sistema de supervisão quando o uso for restrito a verificação de status do sistema.

Para acessar a página WEB da UCP desejada, basta utilizar um navegador padrão (Internet Explorer 7 ou superior, Mozilla Firefox 3.0 ou superior e Google Chrome 8 ou superior) e digitar, na barra de endereço, o endereço IP correspondente a UCP (Ex.: http://192.168.1.1). Inicialmente, será apresentada as informações da UCP, conforme mostra a Figura 7-3:

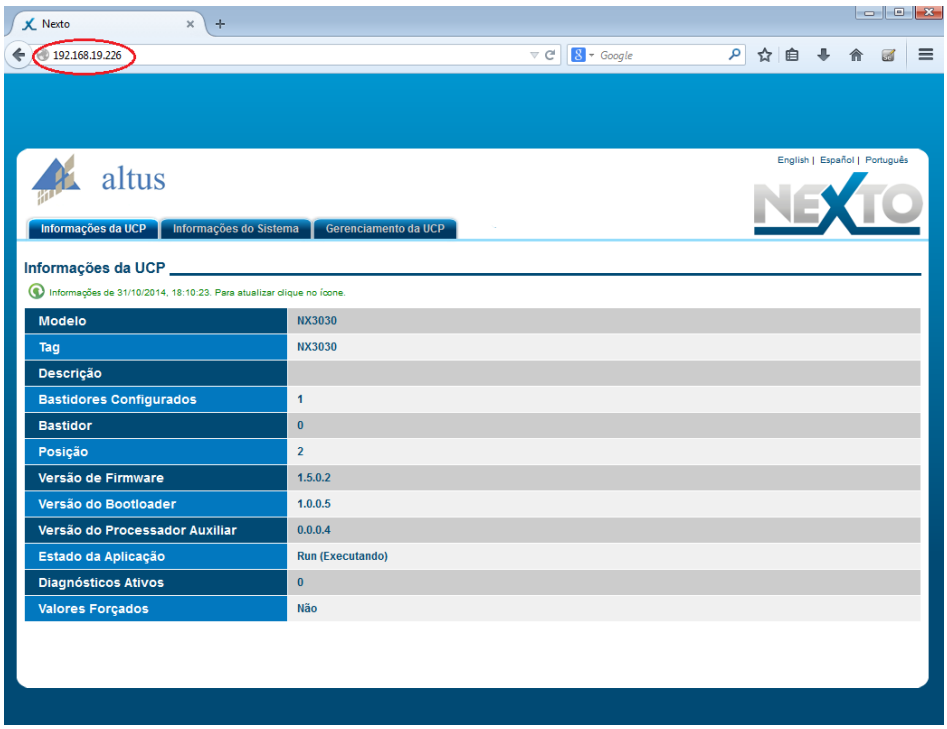

Figura 7-3. Tela Inicial

Também existe a aba "Informações do Sistema", a qual pode ser visualizada através do Rack ou da lista dos módulos presentes (opção do lado direito da tela). Quando não houver nenhuma aplicação na UCP, será exibida nesta página uma configuração com o maior Rack disponível e uma fonte de alimentação padrão, juntamente com a UCP conectada. Quando a visualização pelo Rack é utilizada, os módulos que têm diagnóstico ficam piscando e assumem a cor vermelha, conforme mostra a Figura 7-4. Caso contrário será exibida uma lista com os módulos presentes no sistema, Tags correspondentes e número de diagnósticos ativos:

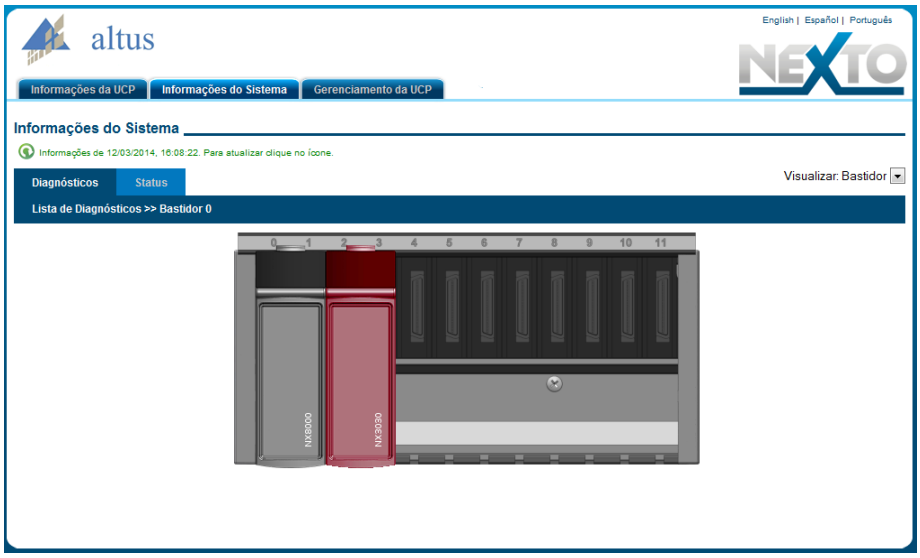

Figura 7-4. Informações do Sistema

Ao clicar no módulo com diagnóstico, no mesmo instante são mostrados os diagnósticos ativos do módulo, conforme mostra a Figura 7-5:

ATENÇÃO:
Quando uma UCP for reiniciada e a aplicação entrar em exceção na partida do sistema, os diagnósticos não estarão válidos. É necessário corrigir o problema que gera a exceção da aplicação para que os diagnósticos sejam atualizados.

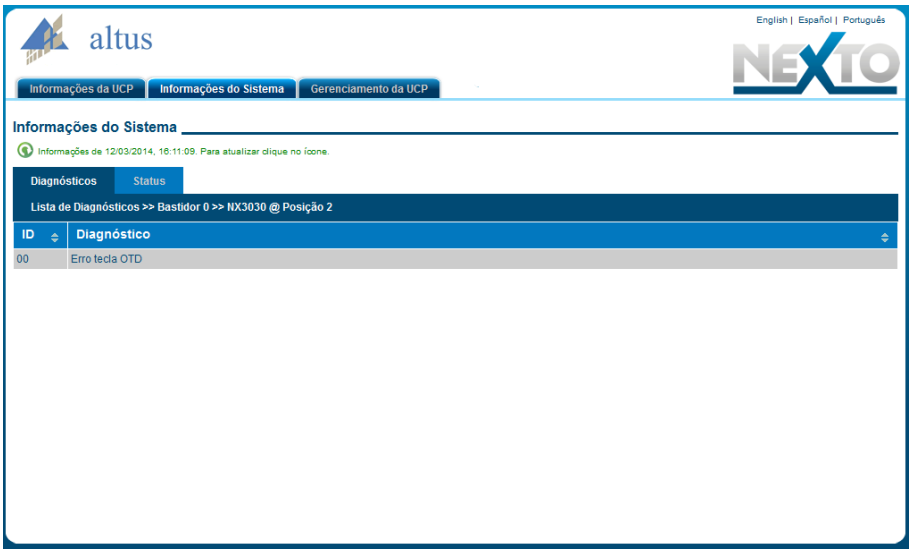

Figura 7-5. Diagnósticos do Sistema

Caso a aba Status seja selecionada, o estado de todos os diagnósticos detalhados é exibido na tela, conforme mostra a Figura 7-6:

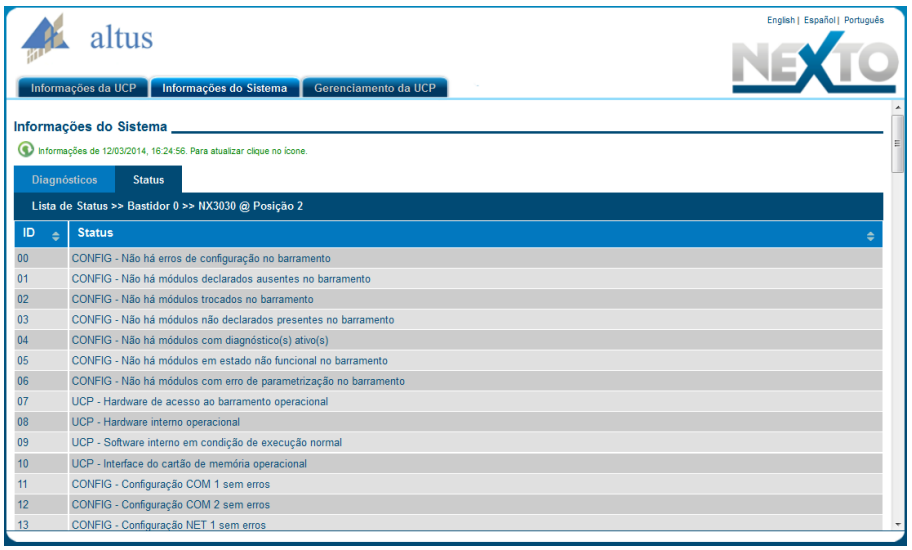
altus
English | Español | Português
NEXTO
Informações da UCP
Informações do Sistema
Gerenciamento da UCP
Informações do Sistema
Informações de 12/03/2014, 16:24:56. Para atualizar clique no ícone.
Diagnósticos
Status
Lista de Status >> Bastidor 0 >> NX3030 @ Posição 2
ID
Status
00 CONFIG - Não há erros de configuração no barramento
01 CONFIG - Não há módulos declarados ausentes no barramento
02 CONFIG - Não há módulos trocados no barramento
03 CONFIG - Não há módulos não declarados presentes no barramento
04 CONFIG - Não há módulos com diagnóstico(s) ativo(s)
05 CONFIG - Não há módulos em estado não funcional no barramento
06 CONFIG - Não há módulos com erro de parametrização no barramento
07 UCP - Hardware de acesso ao barramento operacional
08 UCP - Hardware interno operacional
09 UCP - Software interno em condição de execução normal
10 UCP - Interface do cartão de memória operacional
11 CONFIG - Configuração COM 1 sem erros
12 CONFIG - Configuração COM 2 sem erros
13 CONFIG - Configuração NET 1 sem erros

Figura 7-6. Estado do Sistema

Além disso, o usuário pode optar por três opções de idioma: Português, Inglês e Espanhol. Basta alterar o menu superior direito para o idioma desejado. Na aba Gerenciamento da UCP, existem outros recursos como Atualização de Firmware e SNMP.

A aba Atualização de Firmware é restrita ao usuário, ou seja, apenas para uso interno da Altus. Em casos onde a atualização é realizada remotamente (através de uma conexão de rádio ou satélite por exemplo, a velocidade mínima deste link deve ser de 128Kbps).

Diagnostic Explorer

O Diagnostic Explorer é a inclusão dos diagnósticos via WEB dentro do MasterTool IEC XE, a fim de que o acesso seja mais rápido e objetivo.

O acesso à característica ocorre de duas maneiras:

Acessando a opção "Diagnostic Explorer" na árvore de dispositivos, localizada na esquerda da tela do MasterTool IEC XE, colocando o IP correto no campo indicado na Figura 7-7. Lembrando que para a página de diagnósticos ser exibida, o usuário deverá estar conectado à UCP (capítulo Programação Inicial – Login).

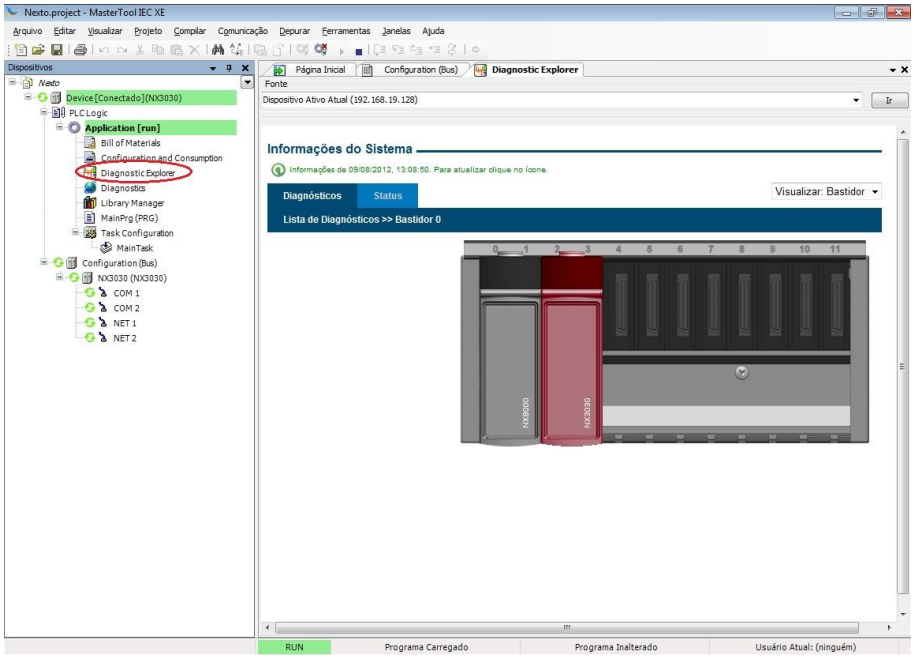

Figura 7-7. Tela do Diagnostic Explorer

- Clicando com o botão direito do mouse sobre o módulo, e selecionando "Diagnósticos", o Diagnostic Explorer será aberto, direcionando para a página de status do respectivo módulo.

Diagnósticos via Variáveis

As UCPs da Série Nexto possuem variáveis para indicação de diagnóstico. Existem estruturas de dados com os diagnósticos de todos os módulos declarados no barramento, mapeadas sobre variáveis de representação direta %Q, e definidas simbolicamente através da diretiva AT, na GLV Diagnostics criada automaticamente pelo MasterTool IEC XE.

A Tabela 7-5 faz um resumo da divisão dos bytes/words de diagnósticos:

Byte	Descrição
0 a 3	Diagnósticos resumidos da UCP.
4 a 558	Diagnósticos detalhados da UCP (NX3010 e NX3004).
4 a 693	Diagnósticos detalhados da UCP (NX3020 e NX3030).

Tabela 7-5. Divisão dos Diagnósticos da UCP

Diagnósticos Resumidos

A Tabela 7-6 mostra o significado de cada bit dos diagnósticos resumidos da UCP:

Variável de Representação Direta	Mensagem de Diagnóstico	Variável DG_Modulo.tSummarized.*	Descrição

Variável	Bit			
-	-	SEM DIAG	-	Não existe diagnóstico ativo.
%QB(n)	0.0	CONFIG. INCOMPATIVEL	bConfigMismatch	TRUE – Existe algum problema de configuração no barramento, como módulo em posição incorreta.
				FALSE – O barramento está configurado corretamente.
	0.1	MODULOS AUSENTES	bAbsentModules	TRUE – Um ou mais módulos declarados estão ausentes.
				FALSE – Todos os módulos estão presentes no barramento.
	0.2	MODULOS TROCADOS	bSwappedModules	TRUE – Dois módulos estão trocados entre si no barramento.
				FALSE – Não há módulos trocados no barramento.
	0.3	MODULOS NÃO DECLARADOS	bNonDeclaredModules	TRUE – Um ou mais módulos presentes no barramento não estão declarados.
				FALSE – Todos os módulos presentes no barramento estão declarados.
	0.4	MODULOS C/ DIAGNOSTICO	bModulesWithDiagnostic	TRUE – Um ou mais módulos de do barramento estão com diagnóstico ativo.
				FALSE – Não existem diagnósticos ativos nos módulos do barramento.
	0.5	MODULOS C/ ERRO FATAL	bModuleFatalError	TRUE – Um ou mais módulos presentes barramento estão em estado não funcional.
				FALSE – Todos os módulos presentes no barramento estão em estado funcional.
	0.6	MODULOS C/ ERRO PARAM.	bModuleParameterError	TRUE – Um ou mais módulos do barramento estão com erro de parametrização.
				FALSE – Todos os módulos estão parametrizados.
	0.7	ERRO NO BARRAMENTO	bWHSBBusError	TRUE – Indicação do mestre que existe falha no barramento WHSB.
				FALSE – O barramento WHSB está em correto funcionamento.
%QB(n+1)	1.0	FALHA DE HARDWARE	bHardwareFailure	TRUE – Falha no hardware da UCP.
				FALSE – O hardware está em correto funcionamento.
	1.1	EXCECAO NO SOFTWARE	bSoftwareException	TRUE – Uma ou mais exceções geradas pelo software.
				FALSE – Não foram geradas exceções no software.
	1.2		bReserved_10	Reservado.
	1.3	ERRO CARTAO DE MEMORIA	bMemoryCardError	TRUE – O cartão de memória está inserido na UCP, porém não está funcionando corretamente.
				FALSE – O cartão de memória está funcionando corretamente.
	1.4	ERRO CONFIG. COM 1	bCOM1ConfigError	TRUE – Ocorreu algum erro durante, ou após, a configuração da interface serial COM 1.
				FALSE – A configuração da interface serial COM 1 está correta.
	1.5	ERRO CONFIG. COM 2	bCOM2ConfigError	TRUE – Ocorreu algum erro durante, ou após, a configuração da interface serial COM 2.
				FALSE – A configuração da interface serial COM 2 está correta.
	1.6	ERRO CONFIG. NET 1	bNET1ConfigError	TRUE – Ocorreu algum erro durante, ou após, a configuração da interface Ethernet NET 1.
				FALSE – A configuração da interface Ethernet NET 1 está correta.
	1.7	ERRO CONFIG. NET 2	bNET2ConfigError	TRUE – Ocorreu algum erro durante, ou após, a configuração da interface Ethernet NET 2.
				FALSE – A configuração da interface Ethernet NET 2 está correta.

%QB(n+2)	2.0	DATA/HORA INVALIDA	bInvalid DateTime	TRUE – A data ou a hora são inválidas.
				FALSE – A data e a hora estão corretas.
	2.1	RUNTIME RESET	bRTSReset	TRUE – O RTS (Runtime System) foi reiniciado pelo menos uma vez. Esse diagnóstico somente é limpo na reinicialização do sistema.
				FALSE – O RTS (Runtime System) está operando normalmente.
	2.2	ERRO TECLA OTD	bOTDSwitchError	TRUE –A tecla ficou travada por mais de 20 s pelo menos uma vez enquanto a UCP esteva energizada. Esse diagnóstico somente é limpo na reinicialização do sistema.
				FALSE – A tecla não está ou ficou travada enquanto a UCP esteve energizada.
	2.3 a 2.7		bReserved_xx	Reservado.
%QB(n+3)	3.0	BASTIDOR AUSENTE	bAbsentRacks	TRUE – Um ou mais bastidores declarados estão ausentes.
				FALSE – Todos os bastidores declarados estão presentes.
	3.1	BASTIDOR DUPLICADO	bDuplicatedRacks	TRUE – Há algum bastidor com o número de identificação duplicado.
				FALSE – Não existem bastidores com o número de identificação duplicado.
	3.2	BASTIDOR INVALIDO	bInvalidRacks	TRUE – Há algum bastidor com o número de identificação inválido.
				FALSE – Não existem bastidores com o número de identificação inválido.
	3.3	BASTIDOR NÃO DECLARADO	bNonDeclaredRacks	TRUE – Há algum bastidor com o número de identificação não declarado.
				FALSE – Não existem bastidores com o número de identificação não declarado.
	3.4	SLOT DUPLICADO	bDuplicatedSlots	TRUE – Há algum endereço de slot duplicado.
				FALSE – Não existem endereços de slot duplicados.
	3.5 a 3.7		bReserved_xx	Reservado.

Tabela 7-6. Tabela dos Diagnósticos Resumidos da UCP

Notas:

Variável de representação direta: O "n" representa o valor configurado na UCP, através do software MasterTool IEC XE, como endereço inicial de diagnósticos.

Diretiva AT: Na descrição das variáveis simbólicas que utilizam a diretiva AT para fazer o mapeamento em variáveis de endereçamento direto, a sintaxe que deve ser colocada antes do diagnóstico resumido desejado é DG_Modulo.tSummarized., sendo a palavra Modulo substituída pela UCP utilizada. Por exemplo, para o diagnóstico de configuração incompatível basta utilizar a seguinte variável, DG_NX3010.tSummarized.bConfigMismatch. A diretiva AT é uma palavra reservada no software programador, sendo que algumas variáveis simbólicas que utilizam essa diretiva servem para indicar os diagnósticos.

Configuração incompatível: O diagnóstico de configuração incompatível é gerado caso um ou mais módulos do barramento não conferirem com o que foi declarado, ou seja, nas condições de módulos ausentes ou diferentes. Os módulos inseridos no barramento que não foram declarados no projeto não são considerados.

Módulos trocados: Se somente dois módulos estiverem trocados entre si no respectivo barramento, então o diagnóstico de troca pode ser identificado. Caso contrário, o problema será tratado como "Configuração Incompatível".

Módulos com erro fatal: Caso o diagnóstico de Módulos com Erro Fatal seja verdadeiro, verificar qual é o módulo que está com problema no barramento e encaminhar para a Assistência Técnica da Altus, pois o mesmo está apresentando falha do hardware.

Módulos com erro de parametrização: Caso o diagnóstico de Módulo com Erro de Parametrização seja verdadeiro, verificar se os módulos do barramento estão configurados corretamente e se as versões de firmware e do software MasterTool IEC XE estão adequadas.

Erro no barramento: Considerado um erro fatal, interrompendo o acesso aos módulos do barramento. Caso o diagnóstico de erro no barramento seja verdadeiro, talvez haja um problema de hardware nas linhas de comunicação do barramento, sendo assim, deve-se entrar em contato com a Assistência Técnica da Altus.

Falha de hardware: Caso o diagnóstico de Falha de Hardware seja verdadeiro, encaminhar a UCP para Assistência Técnica da Altus, pois a mesma apresenta problemas no RTC, processador auxiliar, ou outros recursos de hardware.

Exceção no software: Caso o diagnóstico de exceção no software seja verdadeiro, o usuário deverá verificar a sua aplicação para garantir que a mesma não esteja acessando indevidamente a memória. Se o problema persistir, o setor de Suporte da Altus deverá ser consultado. Os códigos de exceção no software estão descritos após a tabela de diagnósticos detalhados da UCP.

Erro do cartão de memória: Os diagnósticos referentes ao cartão de Memória estão disponíveis somente para as UCPs NX3010, NX3020 e NX3030.

Mensagem de Diagnóstico: As mensagens de diagnóstico podem ser visualizadas através do visor gráfico da UCP através da tecla OTD ou através da WEB na página de diagnósticos da UCP.

Interfaces Seriais: Os diagnósticos referentes à interface COM 2 estão disponíveis somente para as UCPs NX3010, NX3020 e NX3030.

Interfaces Ethernet: Os diagnósticos referentes à interface NET 2 estão disponíveis somente para as UCPs NX3020 e NX3030.

Diagnósticos Detalhados

As tabelas a seguir mostram os diagnósticos detalhados das UCPs da série Nexto, para sua consulta é importante verificar as observações abaixo:

- Visualização das Estruturas de Diagnóstico: As Estruturas de Diagnóstico adicionadas ao projeto podem ser visualizadas acessando o item "Library Manager" na treeview da janela do MasterTool IEC XE. Com isso, é possível visualizar todos os datatypes definidos na estrutura.
- Contadores: Todos os contadores dos diagnósticos da UCP retornam a zero quando o seu valor limite é ultrapassado.
- Variável de representação direta: O "n" representa o valor configurado na UCP, através do software MasterTool IEC XE, como endereço inicial de diagnósticos.
- Diretiva AT: Na descrição das variáveis simbólicas que utilizam a diretiva AT para fazer o mapeamento em variáveis de endereçamento direto, a sintaxe que deve ser colocada antes do diagnóstico resumido desejado é DG_Modulo.tSummarized., sendo a palavra Modulo substituída pela UCP utilizada. Por exemplo, para o diagnóstico de configuração incompatível basta utilizar a seguinte variável, DG_NX3010.tSummarized.bConfigMismatch. A diretiva AT é uma palavra reservada no software programador, sendo que algumas variáveis simbólicas que utilizam essa diretiva servem para indicar os diagnósticos.

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QD(n+4)				DWORD	Target.dwCPUModel	NX3004 = 0x3004 NX3010 = 0x3010 NX3020 = 0x3020 NX3030 = 0x3030
%QB(n+8)				BYTE ARRAY(4)	Target.abyCPUVersion	Versão do firmware.
%QB(n+12)				BYTE ARRAY(4)	Target.abyBootloaderVersion	Versão do bootloader.
%QB(n+16)				BYTE ARRAY(4)	Target.abyAuxprocVersion	Versão do processador auxiliar.

Tabela 7-7. Descrição dos Diagnósticos Detalhados Grupo Target

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QX(n+20).0				BIT	Hardware.bAuxprocFailure	Falha na comunicação entre o processador auxiliar e o processador principal.
%QX(n+20).1				BIT	Hardware.bRTCFailure	O processador principal não está habilitado para comunicar com o RTC (relógio da UCP).
%QX(n+20).2				BIT	Hardware.bThermometerFailure	Falha na comunicação entre o termômetro e o processador principal.
%QX(n+20).3				BIT	Hardware.bLCDFailure	Falha na comunicação entre o visor LCD e o processador principal.

Tabela 7-8. Descrição dos Diagnósticos Detalhados Grupo Hardware

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QW(n+21)				WORD	Exception.wExceptionCode	Código de Exceção gerado pelo RTS, o significado dos códigos pode ser conferido na Tabela 7-10.
%QB(n+23)				BYTE	Exception.byProcessorLoad	Nível, em percentagem (%), de carga no processador.
%QB(n+24)				BYTE	byReserved_0	Reservado.

Tabela 7-9. Descrição dos Diagnósticos Detalhados Grupo Exception

Nota:

Código de exceção: O código de exceção gerado pelo RTS (Runtime System) pode ser consultado abaixo:

Código	Descrição	Código	Descrição
0x0000	Não há exceção.	**0x0051**	Violação de acesso.
0x0010	Tempo do Cão-de-Guarda da tarefa IEC expirado(Cão-de-Guarda de Software).	**0x0052**	Instrução privilegiada.
0x0012	Erro na configuração dos IOs.	**0x0053**	Falha na página.
		0x0054	Overflow da pilha.
0x0013	Erro de Checksum após o download do Programa.	**0x0055**	Disposição inválida.
0x0014	Erro Fieldbus.	**0x0056**	Manobra inválida.

0x0015	Erro na atualização dos IOs.	**0x0057**	Página protegida.
0x0016	Tempo de ciclo(execução) excedido.	**0x0058**	Falha dupla.
0x0017	Atualização da aplicação Online é muito grande.	**0x0059**	OpCode inválido.
0x0018	Referências externas não resolvidas.	**0x0100**	Desalinhamento do tipo de dado.
0x0019	Download foi rejeitado.	**0x0101**	Limite de arrays excedidos.
0x001A	Projeto não carregado, pois as variáveis retentivas não podem ser realocadas.	**0x0102**	Divisão por zero.
0x001B	Projeto não carregado e deletado.	**0x0103**	Overflow.
0x001C	Fora da pilha da memória.	**0x0104**	Não continuável.
0x001D	Memória retentiva corrompida ou não pode ser mapeada.	**0x0105**	Cão-de-guarda na carga do processador de todas as tarefas IEC detectadas.
0x001E	Projeto pode ser carregado, mas causa uma queda posteriormente.	**0x0150**	FPU: Erro não especificado.
0x0021	Target da aplicação de inicialização não corresponde ao target atual.	**0x0151**	FPU: Operando não normal.
0x0022	Erro nas tarefas escalonadas.	**0x0152**	FPU: Divisão por zero.
0x0023	Checksum do arquivo baixado não corresponde.	**0x0153**	FPU: Resultado inexato.
0x0024	Identidade retentiva não corresponde com a atual identidade da aplicação de inicialização.	**0x0154**	FPU: Operação inválida.
0x0025	Falha na configuração da tarefa IEC.	**0x0155**	FPU: Overflow.
0x0026	Aplicação está executando com target errado.	**0x0156**	FPU: Verificação da pilha.
0x0050	Instrução ilegal.	**0x0157**	FPU: Underflow.

Tabela 7-10. Códigos de Exceção RTS

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	**NX3010**	**NX3020**	**NX3030**			
%QB(n+25)				BYTE	RetainInfo.byCPUInitStatus	Status da Inicialização da UCP: 01: Hot start 02: Warm Start 03: Cold Start Obs.: Essas variáveis são reinicializadas em todas as energizações.
%QW(n+26)				WORD	RetainInfo.wCPUColdStartCounter	Contador de Inicializações à frio: Será incrementado somente devido a retirada a quente da UCP no barramento e não devido ao comando de Reset a Frio do MasterTool IEC XE. (0 a 65535)
%QW(n+28)				WORD	RetainInfo.wCPUWarmStartCounter	Contador de Inicializações à quente: Será incrementado somente durante a sequência de energizações do sistema e não devido ao comando de Reset a Quente do MasterTool IEC XE. (0 a 65535).
%QW(n+30)				WORD	RetainInfo.wCPUHotStartCounter	Contador de distúrbios menores do que o tempo de suporte à falhas na alimentação da UCP. (0 a 65535).
%QW(n+32)				WORD	RetainInfo.wRTSResetCounter	Contador de reset efetuados pelo RTS (Runtime System). (0 a 65535).
%QW(n+34)				WORD	RetainInfo.wReserved_0	Reservado.

Tabela 7-11. Descrição dos Diagnósticos Detalhados Grupo RetainInfo

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QX(n+36).0				BIT	Reset.bBrownOutReset	A UCP foi reiniciada devido a uma falha na alimentação na última inicialização.
%QX(n+36).1				BIT	Reset.bWatchdogReset	A UCP foi reiniciada devido ao cão-de-guarda ativo na última inicialização.

Tabela 7-12. Descrição dos Diagnósticos Detalhados Grupo Reset

Nota:

Reset por Brownout: O diagnóstico de reset por brownout somente será verdadeiro quando a alimentação da fonte exceder o limite mínimo exigido nas características técnicas da mesma, mantendo-se com tensão baixa, ou seja, sem sofrer uma interrupção. A UCP irá identificar a queda da alimentação e indicará o diagnóstico de falha na alimentação. Quando a tensão for restabelecida, a UCP será reinicializada automaticamente e indicará o diagnóstico de reset por brownout.

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QX(n+37).0				BIT	Thermometer.bOverTemperatureAlarm	Alarme gerado devido à temperatura interna estar em 85 ºC ou acima de 85 ºC.
%QX(n+37).1				BIT	Thermometer.bUnderTemperatureAlarm	Alarme gerado devido à temperatura interna estar em 0 ºC ou abaixo de 0 ºC.
%QD(n+38)				DINT	Thermometer.diTemperature	Temperatura lida no sensor interno da UCP.

Tabela 7-13. Descrição dos Diagnósticos Detalhados Grupo Thermometer

Nota:

Temperatura: Para a visualização da temperatura diretamente no endereço de memória, deve-se realizar uma conversão, pois o tamanho do dado é DINT e a monitoração é realizada em 4 bytes. Por isso, indica-se a utilização da variável simbólica associada, pois a mesma já fornece o valor final da temperatura.

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QB(n+42)				BYTE	Serial.COM1.byProtocol	Protocolo selecionado na COM 1: 00: Sem protocolo 01: MODBUS RTU Master 02: MODBUS RTU Slave 03: Outro protocolo
%QD(n+43)				DWORD	Serial.COM1.dwRXBytes	Contador de caracteres recebidos através da COM 1 (0 a 4294967295).
%QD(n+47)				DWORD	Serial.COM1.dwTXBytes	Contador de caracteres transmitidos através da COM 1 (0 a 4294967295).
%QW(n+51)				WORD	Serial.COM1.wRXPendingBytes	Número de caracteres pendentes no buffer de leitura na COM 1 (0 a 65535).
%QW(n+53)				WORD	Serial.COM1.wTXPendingBytes	Número de caracteres pendentes no buffer de transmissão na COM 1 (0 a 65535).
%QW(n+55)				WORD	Serial.COM1.wBreakErrorCounter	Contadores de erro da COM 1 (0 a 65535). Esses contadores serão reiniciados nas seguintes condições:
%QW(n+57)				WORD	Serial.COM1.wParityErrorCounter	

%QW(n+59)	WORD	Serial.COM1.wFrameErrorCounter	Energização Configuração da porta serial COM 1 Remoção das filas RX e TX
%QW(n+61)	WORD	Serial.COM1.wRXOverrunCounter	
%QW(n+63)	WORD	Serial.COM1.wReserved_0	Reservado.
%QW(n+65)	WORD	Serial.COM1.wReserved_1	Reservado.

Tabela 7-14. Descrição dos Diagnósticos Detalhados Grupo Serial COM 1

Nota:

Contador de erro de paridade: Quando a paridade da interface serial COM 1 estiver configurada como Sem Paridade, este contador de erros não será incrementado ao receber uma mensagem com paridade diferente. Nesse caso, será indicado erro de frame.

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
NA	%QB(n+67)			BYTE	Serial.COM2.byProtocol	Protocolo selecionado na COM 2: 00: Sem protocolo 01: MODBUS RTU Master 02: MODBUS RTU Slave 03: Outro protocolo
NA	%QD(n+68)			DWORD	Serial.COM2.dwRXBytes	Contador de caracteres recebidos através da COM 2 (0 a 4294967295).
NA	%QD(n+72)			DWORD	Serial.COM2.dwTXBytes	Contador de caracteres transmitidos através da COM 2 (0 a 4294967295).
NA	%QW(n+76)			WORD	Serial.COM2.wRXPendingBytes	Número de caracteres pendentes no buffer de leitura na COM 2 (0 a 65535).
NA	%QW(n+78)			WORD	Serial.COM2.wTXPendingBytes	Número de caracteres pendentes no buffer de transmissão na COM 2 (0 a 65535).
NA	%QW(n+80)			WORD	Serial.COM2.wBreakErrorCounter	Contadores de erro da COM 2 (0 a 65535). Esses contadores serão reiniciados nas seguintes condições: Inicialização Configuração da porta serial COM 2 Remoção das filas RX e TX
NA	%QW(n+82)			WORD	Serial.COM2.wParityErrorCounter	
NA	%QW(n+84)			WORD	Serial.COM2.wFrameErrorCounter	
NA	%QW(n+86)			WORD	Serial.COM2.wRXOverrunCounter	
NA	%QW(n+88)			WORD	Serial.COM2.wReserved_0	Reservado.
NA	%QW(n+90)			WORD	Serial.COM2.wReserved_1	Reservado.

Tabela 7-15. Descrição dos Diagnósticos Detalhados Grupo Serial COM 2

Nota:

Contador de erro de paridade: Quando a paridade da interface serial COM 2 estiver configurada como Sem Paridade, este contador de erros não será incrementado ao receber uma mensagem com paridade diferente. Nesse caso, será indicado erro de frame.

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QX(n+92).0				BIT	Ethernet.NET1.bLinkDown	Indica o estado do link na NET 1.
%QW(n+93)				WORD	Ethernet.NET1.wProtocol	Protocolo selecionado na NET 1:

			00: Sem protocolo
%QX(n+93).0	BIT	Ethernet.NET1.wProtocol.bMODBUS_RTU_ETH_Client	Cliente MODBUS RTU via TCP.
%QX(n+93).1	BIT	Ethernet.NET1.wProtocol.bMODBUS_ETH_Client	Cliente MODBUS TCP.
%QX(n+93).2	BIT	Ethernet.NET1.wProtocol.bMODBUS_RTU_ETH_Server	Servidor MODBUS RTU via TCP.
%QX(n+93).3	BIT	Ethernet.NET1.wProtocol.bMODBUS_ETH_Server	Servidor MODBUS TCP.
%QB(n+95)	STRING(15)	Ethernet.NET1.szIP	Endereço IP NET 1.
%QB(n+111)	STRING(15)	Ethernet.NET1.szMask	Máscara de Subrede NET 1.
%QB(n+127)	STRING(15)	Ethernet.NET1.szGateway	Endereço Gateway NET 1.
%QB(n+143)	STRING(17)	Ethernet.NET1.szMAC	Endereço MAC NET 1.
%QB(n+161)	BYTE ARRAY(4)	Ethernet.NET1.abyIP	Endereço IP NET 1.
%QB(n+165)	BYTE ARRAY(4)	Ethernet.NET1.abyMask	Máscara de Subrede NET 1.
%QB(n+169)	BYTE ARRAY(4)	Ethernet.NET1.abyGateway	Endereço Gateway NET 1.
%QB(n+173)	BYTE ARRAY(6)	Ethernet.NET1.abyMAC	Endereço MAC NET 1.
%QD(n+179)	DWORD	Ethernet.NET1.dwPacketsSent	Contador de pacotes enviados através da porta NET 1 (0 a 4294967295).
%QD(n+183)	DWORD	Ethernet.NET1.dwPacketsReceived	Contador de pacotes recebidos através da porta NET 1 (0 a 4294967295).
%QD(n+187)	DWORD	Ethernet.NET1.dwBytesSent	Contador de bytes enviados através da porta NET 1 (0 a 4294967295).
%QD(n+191)	DWORD	Ethernet.NET1.dwBytesReceived	Contador de bytes recebidos através da porta NET 1 (0 a 4294967295).
%QW(n+195)	WORD	Ethernet.NET1.wTXErrors	Contador de erros de transmissão através da porta NET 1 (0 a 65535).
%QW(n+197)	WORD	Ethernet.NET1.wTXFIFOErrors	Contador de erros no buffer de transmissão através da porta NET 1 (0 a 65535).
%QW(n+199)	WORD	Ethernet.NET1.wTXDropErrors	Contador de perdas de conexão na transmissão através da porta NET 1 (0 a 65535).
%QW(n+201)	WORD	Ethernet.NET1.wTXCollisionErrors	Contador de erros de colisão na transmissão através da porta NET 1 (0 a 65535).
%QW(n+203)	WORD	Ethernet.NET1.wTXCarrierErrors	Contador de erros da portadora na transmissão através da porta NET 1 (0 a 65535).
%QW(n+205)	WORD	Ethernet.NET1.wRXErrors	Contador de erros de recepção através da porta NET 1 (0 a 65535).
%QW(n+207)	WORD	Ethernet.NET1.wRXFIFOErrors	Contador de erros no buffer de recepção através da porta NET 1 (0 a 65535).
%QW(n+209)	WORD	Ethernet.NET1.wRXDropErrors	Contador de perdas de conexão na recepção através da porta NET 1 (0 a 65535).
%QW(n+211)	WORD	Ethernet.NET1.wRXFra	Contador de erros de frame na

		meErrors	recepção através da porta NET 1 (0 a 65535).
%QW(n+213)	WORD	Ethernet.NET1.wMulticast	Contador de pacotes multicast através da porta NET 1 (0 a 65535).
%QW(n+215)	WORD	Ethernet.NET1.wReserved_0	Reservado.
%QW(n+217)	WORD	Ethernet.NET1.wReserved_1	Reservado.

Tabela 7-16. Descrição dos Diagnósticos Detalhados Grupo Ethernet NET1

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
NA		%QX(n+219).0		BIT	Ethernet.NET2.bLinkDown	Indica o estado do link na NET 2.
NA		%QW(n+220)		WORD	Ethernet.NET2.wProtocol	Protocolo selecionado na NET 2: 00: Sem protocolo.
NA		%QX(n+220).0		BIT	Ethernet.NET2.wProtocol.bMODBUS_RTU_ETH_Client	Cliente MODBUS RTU via TCP.
NA		%QX(n+220).1		BIT	Ethernet.NET2.wProtocol.bMODBUS_ETH_Client	Cliente MODBUS TCP.
NA		%QX(n+220).2		BIT	Ethernet.NET2.wProtocol.bMODBUS_RTU_ETH_Server	Servidor MODBUS RTU via TCP.
NA		%QX(n+220).3		BIT	Ethernet.NET2.wProtocol.bMODBUS_ETH_Server	Servidor MODBUS TCP.
NA		%QB(n+222)		STRING(15)	Ethernet.NET2.szIP	Endereço IP NET 2.
NA		%QB(n+238)		STRING(15)	Ethernet.NET2.szMask	Máscara de Subrede NET 2.
NA		%QB(n+254)		STRING(15)	Ethernet.NET2.szGateway	Endereço Gateway NET 2.
NA		%QB(n+270)		STRING(17)	Ethernet.NET2.szMAC	Endereço MAC NET 2.
NA		%QB(n+288)		BYTE ARRAY(4)	Ethernet.NET2.abyIP	Endereço IP NET 2.
NA		%QB(n+292)		BYTE ARRAY(4)	Ethernet.NET2.abyMask	Máscara de Subrede NET 2.
NA		%QB(n+296)		BYTE ARRAY(4)	Ethernet.NET2.abyGateway	Endereço Gateway NET 2.
NA		%QB(n+300)		BYTE ARRAY(6)	Ethernet.NET2.abyMAC	Endereço MAC NET 2.
NA		%QD(n+306)		DWORD	Ethernet.NET2.dwPacketsSent	Contador de pacotes enviados através da porta NET 2 (0 a 4294967295).
NA		%QD(n+310)		DWORD	Ethernet.NET2.dwPacketsReceived	Contador de pacotes recebidos através da porta NET 2 (0 a 4294967295).
NA		%QD(n+314)		DWORD	Ethernet.NET2.dwBytesSent	Contador de bytes enviados através da porta NET 2 (0 a 4294967295).
NA		%QD(n+318)		DWORD	Ethernet.NET2.dwBytesReceived	Contador de bytes recebidos através da porta NET 2 (0 a 4294967295).
NA		%QW(n+322)		WORD	Ethernet.NET2.wTXErrors	Contador de erros de transmissão através da porta NET 2 (0 a 65535).
NA		%QW(n+324)		WORD	Ethernet.NET2.wTXFIFOErrors	Contador de erros no buffer de transmissão através da porta NET 2 (0 a 65535).

NA	%QW(n+326)	WORD	Ethernet.NET2.wTXDropErrors	Contador de perdas de conexão na transmissão através da porta NET 2 (0 a 65535).
NA	%QW(n+328)	WORD	Ethernet.NET2.wTXCollisionErrors	Contador de erros de colisão na transmissão através da porta NET 2 (0 a 65535).
NA	%QW(n+330)	WORD	Ethernet.NET2.wTXCarrierErrors	Contador de erros da portadora na transmissão através da porta NET 2 (0 a 65535).
NA	%QW(n+332)	WORD	Ethernet.NET2.wRXErrors	Contador de erros de recepção através da porta NET 2 (0 a 65535).
NA	%QW(n+334)	WORD	Ethernet.NET2.wRXFIFOErrors	Contador de erros no buffer de recepção através da porta NET 2 (0 a 65535).
NA	%QW(n+336)	WORD	Ethernet.NET2.wRXDropErrors	Contador de perdas de conexão na recepção através da porta NET 2 (0 a 65535).
NA	%QW(n+338)	WORD	Ethernet.NET2.wRXFrameErrors	Contador de erros de frame na recepção através da porta NET 2 (0 a 65535).
NA	%QW(n+340)	WORD	Ethernet.NET2.wMulticast	Contador de pacotes multicast através da porta NET 2 (0 a 65535).
NA	%QW(n+342)	WORD	Ethernet.NET2.wReserved_0	Reservado.
NA	%QW(n+344)	WORD	Ethernet.NET2.wReserved_1	Reservado.

Tabela 7-17. Descrição dos Diagnósticos Detalhados Grupo Ethernet NET2

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QB(n+219)		%QB(n+346)		BYTE	UserFiles.byMounted	Indica se a memória utilizada para gravar arquivos de usuário está apta para receber os dados.
%QD(n+220)		%QD(n+347)		DWORD	UserFiles.dwFreeSpacekB	Espaço livre da memória de arquivos de usuário em Kbytes.
%QD(n+224)		%QD(n+351)		DWORD	UserFiles.dwTotalSizekB	Capacidade de armazenamento da memória de arquivos de usuário em Kbytes.
%QB(n+228)		%QB(n+355)		BYTE	UserFiles.byReserved_0	Reservado.

Tabela 7-18. Descrição dos Diagnósticos Detalhados Grupo UserFiles

Nota:

Partição de Usuário: A partição de usuário é uma área da memória reservada para o armazenamento de dados na UCP. Por exemplo: arquivos com extensão PDF, arquivos com extensão DOC e demais dados.

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QB(n+229)		%QB(n+356)		BYTE	UserLogs.byMounted	Status da memória em que são inseridos os logs de usuário.
%QW(n+230)		%QW(n+357)		WORD	UserLogs.wFreeSpacekB	Espaço livre na memória de logs de usuário em Kbytes.
%QW(n+232)		%QW(n+359)		WORD	UserLogs.wTotalSizekB	Capacidade de armazenamento da memória de logs de usuário em Kbytes.
%QB(n+234)		%QB(n+361)		BYTE	UserLogs.byReserved_0	Reservado.

Tabela 7-19. Descrição dos Diagnósticos Detalhados Grupo UserLogs

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
NA	%QB(n+235)	%QB(n+362)		BYTE	MemoryCard.byMounted	Status do Cartão de Memória: 00: Cartão de memória não montado 01: Cartão de memória inserido e montado
NA	%QX(n+236).0	%QX(n+363).0		BIT	MemoryCard.bMemcardtoCPUEnabled	Nível de Proteção do Cartão de Memória: Leitura de dados do cartão de memória pela UCP autorizada.
NA	%QX(n+236).1	%QX(n+363).1		BIT	MemoryCard.bCPUtoMemcardEnabled	Escrita de dados no cartão de memoria pela UCP autorizada.
NA	%QD(n+237)	%QD(n+364)		DWORD	MemoryCard.dwFreeSpacekB	Espaço livre no Cartão de Memória em Kbytes.
NA	%QD(n+241)	%QD(n+368)		DWORD	MemoryCard.dwTotalSizekB	Capacidade de armazenamento do Cartão de Memória em Kbytes.

Tabela 7-20. Descrição dos Diagnósticos Detalhados Grupo MemoryCard

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QB(n+245)		%QB(n+372)		BYTE	WHSB.byHotSwapAndStartupStatus	Informa a situação anormal no barramento que ocasionou a parada da aplicação, para cada Modo de troca a quente. Consulte a Tabela 7-22 para detalhes das possibilidades.
%QB(n+246)		%QB(n+373)		BYTE	WHSB.byReserved_0	Reservado.
%QD(n+247)		%QD(n+374)		DWORD ARRAY(32)	WHSB.adwRackIOErrorStatus	Identificação de erros em módulos de E/S, individualmente. Para mais informações sobre este diagnóstico consultar a nota abaixo.
%QD(n+375)		%QD(n+502)		DWORD ARRAY(32)	WHSB.adwModulePresenceStatus	Status de presença de módulos declarados de E/S em barramentos, individualmente. Para mais informações sobre este diagnóstico consultar a nota abaixo.
%QB(n+503)		%QB(n+630)		BYTE	WHSB.byWHSBBusErrors	Contador de falhas no barramento WHSB. Esse contador é reinicializado na energização (0 a 255).

Tabela 7-21. Descrição dos Diagnósticos Detalhados Grupo WHSB

Notas:

Diagnóstico de erro dos módulos do barramento: Cada DWORD do Array deste diagnóstico representa um bastidor, cujas posições são representadas pelos bits destas DWORDs. Logo, o Bit-0 da DWORD-0 equivale a posição zero do bastidor com endereço zero. Cada um desses Bits é o resultado de uma operação lógica OU entre os diagnósticos de configuração incompatível (bConfigMismatch), módulos ausentes (bAbsentModules), módulos trocados (bSwappedModules), módulos com erro fatal (bModuleFatalError) e o estado operacional do módulo de uma determinada posição.

Status de presença de módulos: Cada DWORD do Array deste diagnóstico representa um bastidor, cujas posições são representadas pelos bits destas DWORDs. Logo, o Bit-0 da DWORD-0 equivale a posição zero do bastidor com endereço zero. Então, se o módulo estiver presente, este bit será verdadeiro. É importante ressaltar que esse diagnóstico é válido para todos os módulos, exceto fontes de alimentação, UCPs e módulos não declarados, ou seja, não apresentam a presença no barramento em suas respectivas posições (bit permanece em falso).

Situações que ocasionam parada da aplicação: Os códigos das possíveis situações que ocasionam parada da aplicação podem ser consultados abaixo:

Código	Enumerável	Descrição
00	INITIALIZING	Esse estado é apresentado enquanto os demais estados não estiverem prontos.
01	RESET_WATCHDOG	Aplicação em Modo Stop devido ao reset por cão-de-guarda de hardware ou por uma reinicialização do Runtime, quando desabilitada a configuração "Iniciar Aplicação de Usuário após Reset por Cão-de-Guarda".
02	ABSENT_MODULES_HOT_SWAP_DISABLED	Aplicação em Modo Stop devido à ativação do diagnóstico Módulos Ausentes, quando configurada troca a quente desabilitada ou troca a quente desabilitada apenas para módulos declarados.
03	CFG_MISMATCH_HOT_SWAP_DISABLED	Aplicação em Modo Stop devido à ativação do diagnóstico Configuração Incompatível, quando configurada troca a quente desabilitada ou troca a quente desabilitada apenas para módulos declarados.
04	ABSENT_MODULES_HOT_SWAP_STARTUP_CONSISTENCY	Aplicação em Modo Stop devido à ativação do diagnóstico Módulos Ausentes, quando configurada troca a quente com consistência na partida ou troca a quente com consistência na partida somente para os módulos declarados.
05	CFG_MISMATCH_HOT_SWAP_STARTUP_CONSISTENCY	Aplicação em Modo Stop devido à ativação do diagnóstico Configuração Incompatível, quando configurada troca a quente com consistência na partida ou troca a quente com consistência na partida somente para os módulos declarados.
06	APPL_STOP_ALLOWED_TO_RUN	Aplicação em Modo Stop e todas as consistências realizadas com sucesso. Aplicação pode ser colocada no Modo Run.
07	APPL_STOP_MODULES_NOT_READY	Aplicação em Modo Stop e todas as consistências realizadas com sucesso, mas os módulos de E/S não estão aptos para a partida do sistema. Não é possível colocar a Aplicação em Modo Run.
08	APPL_STOP_MODULES_GETTING_READY_TO_RUN	Aplicação em Modo Stop e todas as consistências realizadas com sucesso. Os módulos de E/S estão sendo preparados para a partida do sistema. Não é possível colocar a Aplicação em Modo Run.
09	NORMAL_OPERATING_STATE	Aplicação em Modo Run.
10	MODULE_CONSISTENCY_OK	Uso interno.
11	APPL_STOP_DUE_TO_EXCEPTION	Aplicação em modo Stop pois uma exceção ocorreu na UCP.
12	DUPLICATED_SLOT_HOT_SWAP_DISABLED	Aplicação em Modo Stop devido à ativação do diagnóstico Slots Duplicados, quando configurada troca a quente desabilitada ou troca a quente desabilitada apenas para módulos declarados.
13	DUPLICATED_SLOT_HOT_SWAP_STARTUP_CONSISTENCY	Aplicação em Modo Stop devido à ativação do diagnóstico Slots Duplicados, quando configurada troca a quente com consistência na partida ou troca a quente com consistência na partida somente para os módulos declarados.
14	DUPLICATED_SLOT_HOT_SWAP_ENABLED	Aplicação em Modo Stop devido à ativação do diagnóstico Slots Duplicados, quando configurada troca a quente habilitada sem consistência na partida.
15	NON_DECLARED_MODULE_HOT_SWAP_STARTUP_CONSISTENCY	Aplicação em Modo Stop devido à ativação do diagnóstico Módulos Não Declarados, quando configurada troca a quente com consistência na partida.
16	NON_DECLARED_MODULE_HOT_SWAP_DISABLED	Aplicação em Modo Stop devido à ativação do diagnóstico Módulos Não Declarados, quando configurada troca a quente desabilitada.

Tabela 7-22. Códigos de Situações que Ocasionam Parada da Aplicação

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QB(n+504)		%QB(n+631)		BYTE	Application.byCPUState	Informa o estado de operação da UCP: 01: Aplicação em execução (Modo Run) 03: Aplicação parada (Modo Stop)
%QX(n+505).0		%QX(n+632).0		BIT	Application.bForcedIOs	Existem um ou mais pontos de IO forçados.

Tabela 7-23. Descrição dos Diagnósticos Detalhados Grupo Application

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QX(n+532).0		%QX(n+633).0		BIT	SNTP. bServiceEnabled	Serviço SNTP habilitado.
%QB(n+533)		%QB(n+634)		BYTE	SNTP.byActiveTimeServer	Indica qual servidor está ativo: 00: Nenhum servidor ativo. 01: Servidor primário ativo. 02: Servidor secundário ativo.
%QW(n+534)		%QW(n+635)		WORD	SNTP.wPrimaryServerDownCount	Contador de vezes que o servidor primário esteve indisponível (0 a 65535).
%QW(n+536)		%QW(n+637)		WORD	SNTP.wSecondaryServerDownCount	Contador de vezes que o servidor secundário esteve indisponível (0 a 65535).
%QD(n+538)		%QD(n+639)		DWORD	SNTP.dwRTCTimeUpdatedCount	Contador de vezes que o RTC foi atualizado pelo serviço SNTP (0 a 4294967295).
%QB(n+542)		%QB(n+643)		BYTE	SNTP.byLastUpdateSuccessfull	Indica status da última atualização: 00: Não foi atualizado. 01: Última atualização falhou. 02: Última atualização teve sucesso.
%QB(n+543)		%QB(n+644)		BYTE	SNTP.byLastUpdateTimeServer	Indica qual servidor foi utilizado na última atualização: 00: Nenhuma atualização. 01: Servidor primário. 02: Servidor secundário.
%QB(n+544)		%QB(n+645)		BYTE	SNTP.sLastUpdateTime.byDayOfMonth	Dia da última atualização do RTC.
%QB(n+545)		%QB(n+646)		BYTE	SNTP.sLastUpdateTime.byMonth	Mês da última atualização do RTC.
%QW(n+546)		%QW(n+647)		WORD	SNTP.sLastUpdateTime.wYear	Ano da última atualização do RTC.
%QB(n+548)		%QB(n+649)		BYTE	SNTP.sLastUpdateTime.byHours	Hora da última atualização do RTC.
%QB(n+549)		%QB(n+650)		BYTE	SNTP.sLastUpdateTime.byMinutes	Minuto da última atualização do RTC.
%QB(n+550)		%QB(n+651)		BYTE	SNTP.bReservedAlign	Reservado para alinhamento.
%QB(n+551)		%QB(n+652)		BYTE	SNTP.sLastUpdateTime.bySeconds	Segundo da última atualização do RTC.
%QW(n+552)		%QW(n+653)		WORD	SNTP.sLastUpdateTime.wMilliseconds	Milissegundo da última atualização do RTC.
%QW(n+554)		%QW(n+655)		WORD	SNTP.wReserved_0	Reservado.
%QW(n+556)		%QW(n+657)		WORD	SNTP.wReserved_1	Reservado.

Tabela 7-24. Descrição dos Diagnósticos Detalhados Grupo SNTP

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
NA		%QX(n+659).0		BIT	SOE[1].bConnectionStatus	Status de conexão do cliente 01
NA		%QX(n+659).1		BIT	SOE[1].bOverflowStatus	Status da fila de eventos do cliente 01: FALSE - Não houve estouro TRUE - Limite da fila excedido
NA		%QB(n+660)		BYTE	SOE[1].byReserved_0	Reservado
NA		%QW(n+661)		WORD	SOE[1].wEventsCounter	Contador de eventos na fila do cliente 01
NA		%QX(n+663).0		BIT	SOE[2].bConnectionStatus	Status de conexão do cliente 02
NA		%QX(n+663).1		BIT	SOE[2].bOverflowStatus	Status da fila de eventos do cliente 02: FALSE - Não houve estouro TRUE - Limite da fila excedido
NA		%QB(n+664)		BYTE	SOE[2].byReserved_0	Reservado.
NA		%QW(n+665)		WORD	SOE[2].wEventsCounter	Contador de eventos na fila do cliente 02.

Tabela 7-25. Descrição dos Diagnósticos Detalhados Grupo SOE

Notas:

Sincronismo dos diagnósticos do grupo SOE em sistema operando com redundância de Half-Cluster: Quando um projeto é configurado com redundância de Half-Cluster os diagnósticos do grupo SOE não são sincronizados entre os dois Half-Clusters.

Atualização dos diagnósticos do grupo SOE na transição para o estado ativo: Quando um Half-Cluster passa do estado Reserva para o estado Ativo os diagnósticos do grupo SOE passam a ser atualizados a partir do terceiro ciclo.

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QD(n+506)		%QD(n+667)		DWORD	Rack.dwAbsentRacks	Cada bit representa um numero de identificação de um bastidor, se algum bit for TRUE, isso significa que o bastidor, com o referido número de identificação, está ausente.
%QD(n+510)		%QD(n+671)		DWORD	Rack.dwDuplicatedRacks	Cada bit representa um número de identificação de um bastidor, se algum bit for TRUE, isso significa que mais de um bastidor está configurado com o mesmo número de identificação.
%QD(n+514)		%QD(n+675)		DWORD	Rack.dwNonDeclaredRacks	Cada bit representa um numero de identificação de um bastidor, se algum bit for TRUE, significa que há algum bastidor configurado com um número de identificação que não está declarado no projeto.
%QW(n+518)		%QW(n+679)		WORD	Rack.wReserved_0	Reservado.

Tabela 7-26. Descrição dos Diagnósticos Detalhados Grupo Rack

Variável de Representação Direta UCP				Tamanho	Variável AT DG_Modulo.tDetailed.*	Descrição
NX3004	NX3010	NX3020	NX3030			
%QD(n+520)		%QD(n+681)		DWORD	ApplicationInfo.dwApplicationCRC	CRC de 32 bits da aplicação. Quando a aplicação é modificada e enviada para a UCP, um novo

				CRC é calculado.
%QD(n+524)	%QD(n+685)	DWORD	ApplicationInfo.dwReserved_0	Reservado.
%QD(n+528)	%QD(n+689)	DWORD	ApplicationInfo.dwReserved_0	Reservado.

Tabela 7-27. Descrição dos Diagnósticos Detalhados Grupo ApplicationInfo

Diagnósticos via Blocos Funcionais

Os blocos funcionais proporcionam a visualização de alguns parâmetros que não podem ser acessados de outra maneira. As três funções sobre diagnósticos avançados estão localizadas na biblioteca NextoStandard e estão descritas abaixo.

GetTaskInfo

Essa função retorna informações sobre uma tarefa de uma determinada aplicação.

Figura 7-8. Função GetTaskInfo

Abaixo, são descritos os parâmetros que devem ser repassados à função para que ela retorne as informações da aplicação.

Parâmetros de entrada	Tipo	Descrição
psAppName	POINTER TO STRING	Nome da aplicação.
psTaskName	POINTER TO STRING	Nome da tarefa.
pstTaskInfo	POINTER TO stTaskInfo	Ponteiro para receber informações da tarefa.

Tabela 7-28. Parâmetros de Entrada

Os dados que a função retorna, através do ponteiro informado nos parâmetros de entrada, são os descritos na Tabela 7-29.

Parâmetros retornados	Tamanho	Descrição
dwCurScanTime	DWORD	Tempo de ciclo (execução) da tarefa com 1µs de resolução.
dwMinScanTime	DWORD	Tempo mínimo de ciclo da tarefa com 1µs de resolução.
dwMaxScanTime	DWORD	Tempo máximo de ciclo da tarefa com 1µs de resolução.
dwAvgScanTime	DWORD	Tempo médio de ciclo da tarefa com 1µs de resolução.
dwLimitMaxScan	DWORD	Tempo máximo de ciclo da tarefa antes de ocorrer o cão-de-guarda.
dwIECCycleCount	DWORD	Contador de ciclos IEC decorridos.

Tabela 7-29. Parâmetros Retornados

Possíveis ERRORCODE:

- NoError: execução com sucesso;
- TaskNotPresent: a tarefa desejada não existe.

Exemplo de utilização em Linguagem ST:

```
PROGRAM MainPrg
VAR

    sAppName : STRING;
```

```
    psAppName : POINTER TO STRING;
    sTaskName : STRING;
    psTaskName : POINTER TO STRING;
    pstTaskInfo : POINTER TO stTaskInfo;
    TaskInfo : stTaskInfo;
    Info : ERRORCODE;
END_VAR
//ENTRADAS:
sAppName := 'Application';       //Variável recebe o nome da aplicação.
psAppName := ADR(sAppName);      //Ponteiro com o nome da aplicação.
sTaskName := 'MainTask';         //Variável recebe o nome da tarefa.
psTaskName := ADR(sTaskName);    //Ponteiro com o nome da tarefa.
pstTaskInfo := ADR(TaskInfo);    //Ponteiro que irá receber as informações
da tarefa.
//FUNÇÃO:
//Chamada da função.
Info := GetTaskInfo (psAppName, psTaskName, pstTaskInfo);
//Variável 'Info' recebe possíveis erros da função.
```

Visor Gráfico

O visor gráfico disponível nas UCPs da Série Nexto é uma importante ferramenta para o controle de processo, pois através dele podem ser reconhecidas as possíveis condições de erro, presença de componentes ou de diagnósticos ativos. Além disso, é através do visor gráfico que todos os diagnósticos, inclusive dos módulos de E/S, são exibidos ao usuário. Para maiores detalhes sobre a utilização do botão de diagnósticos e da visualização dos mesmos, consultar a seção One Touch Diag.

Abaixo, na Figura 7-9, é possível visualizar todos os caracteres disponíveis no visor gráfico da UCP Nexto e, a seguir, os seus respectivos significados.

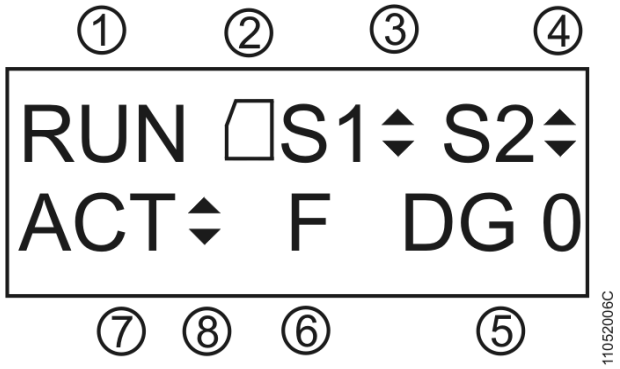

Figura 7-9. Tela de Status da UCP

Legenda:

1. Indicação do estado de operação da UCP. Caso a aplicação da UCP esteja em execução, o estado será Run. Caso a aplicação da UCP esteja parada, o estado será STOP e quando estiver parado em marcas de depuração da aplicação o estado será BRKP. Para Maiores detalhes, consultar a seção Estados de Operação da UCP.
2. Indicação da presença do Cartão de Memória. Para maiores detalhes sobre a instalação do cartão, consultar o capítulo Instalação - Instalação do Cartão de Memória.
3. Indicação de Tráfego na COM 1. A seta para cima (▲) indica transmissão de dados e a seta para baixo (▼) indica recepção de dados. Para maiores informações sobre a interface COM 1, consultar seção Interfaces Seriais.
4. Indicação de Tráfego na COM 2. A seta para cima (▲) indica transmissão de dados e a seta para baixo (▼) indica recepção de dados. Para maiores informações sobre a interface COM 2, consultar seção Interfaces Seriais.

5. Indicação da quantidade de diagnósticos ativos na UCP. Caso o número mostrado seja diferente de 0 (zero), existem diagnósticos ativos na UCP. Para maiores detalhes sobre a visualização dos mesmos no visor gráfico da UCP, através do botão de diagnósticos, consultar seção One Touch Diag.
6. Indicação de variáveis forçadas na UCP. Caso o caractere F esteja exibido no visor gráfico, alguma variável está sendo forçada pelo usuário, seja ela variável simbólica ou variável de representação direta mapeada em uma variável simbólica. Para maiores detalhes sobre forçamento de variáveis, consultar seção Modo Run.
7. Identificação do estado da redundância na UCP (mensagem válida somente na NX3030 em modo redundante). Caso a UCP seja o CP Ativo, a informação ACT será apresentada. Os demais estados possíveis são NCF (Não Configurado), STR (Inicializando), INA (Inativo) e SBY (Reserva).
8. Indicação de que está sendo executada a sincronização de projeto. A seta para cima (▲) indica transmissão de dados do projeto e a seta para baixo (▼) indica recepção de dados de um projeto. Para maiores informações sobre a sincronização de projetos, consultar seção Sincronização de Projetos.

Além dos caracteres descritos acima, as UCPs Nexto poderão apresentar algumas mensagens no visor gráfico, correspondentes a algum processo que está sendo executado no momento.

A Tabela 7-30 mostra as mensagens e suas respectivas descrições:

Mensagem	Descrição
FORMATANDO...	Indica que a UCP está formatando o cartão de memória.
ERRO NA FORMATACAO	Indica que ocorreu um erro durante a formatação do cartão de memória pela UCP.
FORMATO INCORR.	Indica que o formato do cartão de memória está incorreto.
SENHA INCORRETA	Indica que a senha digitada não confere com a senha configurada.
TRANSFERINDO...	Indica que o projeto está sendo transferido.
ERRO NA TRANSFER.	Indica que ocorreu um erro na transferência do projeto, ocasionado por algum problema no cartão de memória ou a remoção do mesmo durante a transferência.
TRANSFER. COMPLETA	Indica que a transferência foi completada com êxito.
TIMEOUT TRANSFER.	Indica que ocorreu um time-out (expirou tempo de comunicação) durante a transferência do projeto.
MODELO UCP INVALIDO	Indica que o modelo da UCP é diferente do configurado no projeto que está no cartão de memória.
ERSAO UCP INVALIDA	Indica que a versão da UCP é diferente do configurado no projeto que está no cartão de memória.
APLICACAO CORROMPIDA	Indica que a aplicação que está no cartão de memória está corrompida.
APLICACAO INEXISTENTE	Indica que não existe aplicação dentro do cartão de memória para ser transferida à UCP.
CRC INEXISTENTE	Indica que o CRC da aplicação não existe.
MCF INEXISTENTE	Indica que não existe o arquivo MCF dentro do cartão de memória.
SEM TAG	Não existe Tag configurada para a UCP no MasterTool IEC XE.
SEM DESC	Não existe Descrição configurada para a UCP no MasterTool IEC XE.
ERRO MSG.	Indica que há erro(s) na(s) mensagem(s) de diagnóstico(s) do(s) módulo(s) requisitado(s).
SEM ASSINATURA	Indica que o produto apresentou um problema inesperado. Entre em contato com o setor de Suporte da Altus.
ERRO NA APL. REINICIANDO	Indica que ocorreu um erro na aplicação e o Runtime está reiniciando a aplicação.
APL. NAO CARREGADA	Indica que o Runtime não irá carregar a aplicação.
CARREGANDO APL.	Indica que o Runtime irá carregar a aplicação.
POSICAO INCORRETA	Indica que a UCP está em uma posição incorreta no rack.
ERRO FATAL	Indica que há problemas graves na inicialização da UCP como por exemplo, as partições da UCP não foram devidamente montadas. Entre em contado com o setor de suporte da Altus.

HW-SW INCOMPATIVEL	Indica que o software e hardware da UCP não são compatíveis pois o produto apresentou algum problema inesperado. Entre em contato com o setor de suporte da Altus.
ATUALIZANDO FIRMWARE	Indica que o firmware está sendo atualizado na UCP.
RECEBENDO FIRMWARE	Indica que o arquivo de atualização está sendo transferido para a UCP.
ATUALIZADO	Exibe a versão de firmware atualizada na UCP.
ERRO NA ATUALIZACAO	Indica que houve algum erro durante a atualização de firmware da UCP, ocasionado por falha na comunicação ou problemas de configuração.
REINICIANDO SISTEMA...	Indica que a UCP está sendo reiniciada para que as atualizações tenham efeito.

Tabela 7-30. Outras Mensagens do Visor Gráfico

Log de Sistema

O Log de Sistema é um recurso disponível no programador MasterTool IEC XE. É uma importante ferramenta de registro de processo, pois através dele é possível localizar eventos na UCP que podem indicar condições de erro, presença de componentes ou de diagnósticos ativos. Tais eventos podem ser visualizados em ordem cronológica com uma resolução de milissegundos, com uma capacidade de armazenamento de até mil entradas de log armazenadas na memória interna do CP e não podem ser removidos.

Para acessar esses Logs, basta ir na Árvore de Dispositivos e dar um duplo-click em Device, e em seguida ir na aba Log, onde podem ser visualizadas as centenas de operações como: ciclo máximo das tarefas, acesso de usuários, download da aplicação, alteração online, download e upload da aplicação, forçamento das variáveis, sincronização da aplicação em CPs redundantes, atualização de firmware entre outros eventos e ações.

Para que os Logs possam ser visualizados, basta estar conectado a uma UCP (Caminho Ativo selecionado) e clicar em . Quando este botão é pressionado os Logs serão exibidos e a atualização deles é feita automaticamente. Quando o botão não está pressionado os Logs serão mantidos na tela, mas não serão atualizados. Ou seja, este botão tem dois estados, sendo que um estado mantém os logs sendo atualizados e no outro estado a atualização está desabilitada. Para deixar de exibir os Logs, basta clicar em .

É possível filtrar os Logs em 4 tipos diferentes: Advertência(s), Erro(s), Exceção (s),Informação (s).

Outra maneira de filtrar as mensagens exibidas para o usuário é selecionar qual o componente queremos visualizar.

A *Marca de Tempo* da aba de Log é apresentada pela ferramenta MasterTool a partir de informações fornecidas pelo dispositivo (UCP). A ferramenta MasterTool pode apresentar a *Marca de Tempo* no horário local (do computador) ou no horário UTC, se marcada a opção *Tempo UTC*.

ATENÇÃO:
Se o horário ou o parâmetro de fuso horário do dispositivo estiverem incorretos, a *Marca de Tempo* apresentada pela ferramenta MasterTool não corresponderá ao tempo esperado.

Para maiores informações sobre os Logs de Sistema, favor consultar o Manual de Utilização do MasterTool IEC XE - MU299048 e os subcapítulos Relógio RTC e Sincronização de Tempo deste manual.

ATENÇÃO:
Os logs de sistema das UCPs da série Nexto, a partir da versão de firmware 1.4.0.33, são recarregados no caso de uma reinicialização da UCP ou por uma reinicialização do Runtime System, isto é, será possível visualizar os logs mais antigos quando uma dessas condições ocorrer.

Não Carregar a Aplicação na Inicialização

Caso seja necessário, o usuário pode optar por não carregar uma aplicação já existente na UCP durante sua inicialização. Para isto, basta energizar a UCP com o botão de diagnóstico pressionado e mantê-lo assim até ser exibida a mensagem na tela "APL. NÃO CARREGADA". Após a inicialização da UCP e a exibição no visor da mensagem avisando que a aplicação não será carregada a UCP estará também em Modo Stop. Caso seja feito um login, o software MasterTool IEC XE indicará que não existe nenhuma aplicação na UCP. Para voltar a carregar a aplicação, a UCP deve ser reinicializada, ou um novo download da aplicação deve ser feito.

Falha na Alimentação

A fonte de alimentação da Série Nexto (NX8000) possui um sistema de detecção de falha, de acordo com os níveis definidos em suas características técnicas (consultar Características Técnicas do Módulo de Alimentação 30W 24Vdc – CT114200). Existem duas formas de diagnosticar a falha.

1 – Caso a fonte NX8000 seja energizada com tensão inferior ao limite mínimo exigido, será gerado um diagnóstico de falha de alimentação, o qual será reconhecido pela UCP e a mensagem "FALHA NA ALIMENTAÇÃO" será exibida em seu visor. Quando a alimentação estiver dentro dos limites estabelecidos, a UCP irá reconhecer e, automaticamente, será reiniciada com a aplicação do usuário. O diagnóstico ainda estará ativo para mostrar ao usuário que na última inicialização ocorreu falha na alimentação.

2 – Caso a fonte NX8000 tenha uma queda de tensão para um valor inferior ao limite mínimo exigido e retornar para um valor acima em até 10 ms, a falha de alimentação não será reconhecida pela UCP e não será gerado o diagnóstico, pois o sistema se mantém intacto durante esse tempo. Porém, se a queda de tensão for superior a 10 ms, a mensagem "FALHA NA ALIMENTAÇÃO" será exibida no visor da UCP e o diagnóstico será ativado.

Figura 7-10. Mensagem de Falha na Alimentação

O usuário pode alterar o valor da variável atribuída à falha na alimentação para FALSE durante a execução do aplicativo, facilitando a verificação e tratamento deste diagnóstico.

O diagnóstico de FALHA NA ALIMENTAÇÃO já está previamente mapeado em uma região específica de memória, definida como Diagnósticos Detalhados da UCP. Dessa forma, basta utilizá-lo como uma variável global. O nome da variável está descrita na lista de diagnósticos detalhados da UCP no capítulo Diagnósticos via Variáveis.

Problemas mais Comuns

Se, ao energizar a UCP, a mesma não entra em funcionamento, os seguintes itens devem ser verificados:

- A temperatura ambiente está dentro da faixa suportada pelos equipamentos?
- A fonte de alimentação do bastidor está sendo alimentada com a tensão correta?
- A fonte de alimentação é o módulo mais a esquerda no bastidor (bastidor sendo visto de frente), e a sua direita é seguido pela UCP da Série Nexto?
- Os equipamentos da rede, como hubs, switches ou roteadores, estão alimentados, interligados, configurados e funcionando corretamente?
- O cabo de rede Ethernet está devidamente conectado à porta NET 1 ou NET 2 da UCP Nexto e ao equipamento de rede?
- A UCP da Série Nexto está ligada, em modo de execução (Run) e sem diagnósticos relacionados à hardware?

Se a UCP Nexto indica o estado execução (Run), mas não responde às comunicações solicitadas, sejam elas pelo MasterTool IEC XE ou através de protocolos, os seguintes itens devem ser verificados:

- A configuração dos parâmetros Ethernet da UCP estão corretos?
- O respectivo protocolo de comunicação está configurado corretamente na UCP?
- As variáveis que habilitam as relações MODBUS estão devidamente habilitadas?

Se nenhum problema for identificado, consulte o Suporte a Clientes Altus.

Solução de Problemas

A Tabela 7-31 mostra os sintomas de alguns problemas com suas possíveis causas e soluções. Se o problema persistir, entre em contato com o Suporte Técnico da Altus.

Sintoma	Possível Causa	Solução
Não liga	Falta de alimentação ou alimentado incorretamente.	Verificar se a UCP está conectada corretamente no bastidor. Desenergizar e retirar todos os módulos do barramento, menos à fonte de alimentação e a UCP. Energizar o barramento e verificar o funcionamento da fonte de alimentação, tanto a externa como a fonte conectada ao barramento. Verificar se a tensão de alimentação chega ao borne da fonte de alimentação Nexto, e também se a polarização está correta.
Visor Gráfico da UCP apresenta a mensagem WRONG SLOT	UCP na Posição Incorreta.	No caso dos Modelos NX3010, NX3020 e NX3030, a UCP deve ocupar os slots 2 e 3 do rack 0. Colocá-la na posição correta. A UCP NX3004 deve ocupar os slots 0 e 1 do rack 0. Colocá-la na posição correta, desligá-la e ligá-la novamente.
Não comunica	Mau contato ou mal configurado.	Verificar todas as conexões dos cabos de comunicação. Verificar as configurações das interfaces seriais e Ethernet no software MasterTool IEC XE.
Não reconhece o cartão de memória	Mal conectado ou desmontado.	Verificar se o cartão de memória está conectado corretamente no compartimento. Verificar se o cartão de memória foi colocado do lado correto, conforme a indicação no painel frontal da UCP. Verificar se o cartão de memória não foi desmontado através da tecla MS, localizado no painel frontal, visualizando a indicação no visor gráfico da UCP.

Tabela 7-31. Tabela de Solução de Problemas

Manutenção Preventiva

- Deve-se verificar, a cada ano, se os cabos de interligação estão com as conexões firmes, sem depósitos de poeira, principalmente os dispositivos de proteção.
- Em ambientes sujeitos a contaminação excessiva, deve-se limpar periodicamente o equipamento, retirando resíduos, poeira, etc.
- Os diodos TVS utilizados para a proteção contra transientes causados por descargas atmosféricas devem ser inspecionados periodicamente, pois podem estar danificados ou destruídos caso a energia absorvida esteja acima do limite. Em muitos casos, a falha pode não ser evidente ou facilmente visível. Em aplicações críticas, é recomendável a substituição periódica dos diodos TVS, mesmo os que não apresentarem sinais visíveis de falha.
- Aperto e limpeza do barramento a cada 6 meses.
- Para maiores informações, consultar o Manual de Utilização Série Nexto – MU214000.

8. Glossário

Barramento	Conjunto de sinais elétricos agrupados logicamente com a função de transferir informação e controle entre diferentes elementos de um subsistema.
Baud rate	Taxa com que os bits de informação são transmitidos através de uma interface serial ou rede de comunicação (medido em bits/segundo).
Bit	Unidade básica de informação, podendo estar no estado 0 ou 1.
Broadcast	Disseminação simultânea de informação a todos os nós interligados a uma rede de comunicação.
Byte	Unidade de informação composta por oito bits.
Canal serial	Interface de um equipamento que transfere dados no modo serial.
Cão-de-guarda de Hardware	Circuito eletrônico destinado a verificar a integridade do funcionamento de um equipamento.
Controlador programável	Também chamado de CP. Equipamento que realiza controle sob o comando de um programa aplicativo. É composto de uma UCP, uma fonte de alimentação e uma estrutura de E/S.
CP	Veja controlador programável.
Diretiva AT	Palavras reservadas no software programador, sendo utilizadas para mapear variáveis simbólicas em variáveis de endereçamento direto.
DG	Sigla usada para indicar diagnóstico nos LEDs
Diagnóstico	Procedimento utilizado para detectar e isolar falhas. É também o conjunto de dados usados para tal determinação, que serve para a análise e correção de problemas.
Download	Carga de programa ou configuração no CP.
E/S	Veja entrada/saída.
EIA RS-485	Padrão industrial (nível físico) para comunicação de dados.
Endereço de módulo	Endereço pelo qual o CP realiza acessos a um determinado módulo de E/S.
Entrada/saída	Também chamado de E/S. Dispositivos de E/S de dados de um sistema. No caso de CPs, correspondem tipicamente a módulos digitais ou analógicos de entrada ou saída que monitoram ou acionam o dispositivo controlado.
Escravo	Equipamento ligado a uma rede de comunicação que só transmite dados se for solicitado por outro equipamento denominado mestre.
Frame	Uma unidade de informação transmitida na rede.
Full Duplex	Indica que os dispositivos podem realizar comunicações transmitindo/recebendo dados, em ambos os sentidos simultaneamente, ou seja, podem transmitir e receber ao mesmo tempo.
Gateway	Equipamento para a conexão de duas redes de comunicação com diferentes protocolos.
GVL	Global Variable List, objeto onde são declaradas as variáveis globais utilizadas na aplicação.
Half Duplex	Indica que os dispositivos podem realizar comunicações transmitindo/recebendo dados, mas somente em um sentido de cada vez, ou seja, podem transmitir ou receber dados.
Hardware	Equipamentos físicos usados em processamento de dados onde normalmente são executados programas (software).
HSDN	High Speed Deterministic Network. Rede determinística de alto desempenho.
IEC	Sigla para International Electrotechnical Commission, ou Comissão Eletrotécnica Internacional, é um órgão internacional de normalização que prepara e publica normas internacionais no âmbito elétrico, eletrônico e de tecnologias relacionadas.
IEC 61131	Norma genérica para operação e utilização de CPs. Antiga IEC 1131.
IEC-61131-3	Terceira parte da norma genérica para operação e utilização de CPs, IEC61131.
Interface	Dispositivo que adapta elétrica e/ou logicamente a transferência de sinais entre dois equipamentos.
Interrupção	Evento com atendimento prioritário que temporariamente suspende a execução de um programa e desvia para uma rotina de atendimento específica
Kbytes	Unidade representativa de quantidade de memória. Representa 1024 bytes.
LED	Sigla para light emitting diode. É um tipo de diodo semicondutor que emite luz quando estimulado por eletricidade. Utilizado como indicador luminoso.
Linguagem de programação	Um conjunto de regras e convenções utilizado para a elaboração de um programa.
Lógica	Matriz gráfica onde são inseridas as instruções de linguagem de um diagrama de relés ou blocos funcionais que compõem uma aplicação. Um conjunto de lógicas ordenadas sequencialmente constitui uma unidade de organização de programa, ou seja, uma subdivisão da aplicação.
MasterTool IEC XE	Identifica o programa Altus para microcomputador, executável em ambiente Windows®, que permite o desenvolvimento de aplicativos para os CPs da série Nexto. Ao longo do manual, este programa é referido pela própria sigla ou como programador MasterTool IEC XE.
Menu	Conjunto de opções disponíveis e exibidas por um programa no vídeo/visor e que podem ser selecionadas pelo usuário a fim de ativar ou executar uma determinada tarefa.
Mestre	Equipamento ligado a uma rede de comunicação de onde se originam solicitações de comandos para

	outros equipamentos da rede.
Módulo (referindo-se a hardware)	Elemento básico de um sistema completo que possui funções bem definidas. Normalmente é ligado ao sistema por conectores, podendo ser facilmente substituído.
Módulo (referindo-se a software)	Parte de um programa aplicativo capaz de realizar uma função específica. Pode ser executado independentemente ou em conjunto com outros módulos, trocando informações através da passagem de parâmetros.
Módulo de E/S	Módulo pertencente ao subsistema de entradas e saídas.
Nó	Qualquer estação de uma rede com capacidade de comunicação utilizando um protocolo estabelecido.
Operandos	Elementos sobre os quais as instruções atuam. Podem representar constantes, variáveis ou um conjunto de variáveis.
Protocolo	Regras de procedimentos e formatos convencionais que, mediante sinais de controle, permitem o estabelecimento de uma transmissão de dados e a recuperação de erros entre equipamentos.
PLC	Sigla para Programmable Logic Controller. É a abreviatura de controlador programável em inglês.
POU	Program Organization Unit, ou Unidade de Organização de Programa, é uma subdivisão do programa aplicativo que pode ser escrito em qualquer uma das linguagens disponíveis.
Programa aplicativo	É o programa carregado em um CP, que determina o funcionamento de uma máquina ou processo.
Rede de comunicação	Conjunto de equipamentos (nós) interconectados por canais de comunicação.
Rede de comunicação determinística	Rede de comunicação onde à transmissão e a recepção de informações entre os diversos nós é garantida com um tempo máximo conhecido.
Rede de comunicação mestre-escravo	Rede de comunicação onde às transferências de informações são iniciadas somente a partir de um único nó (mestre da rede) ligado ao barramento de dados. Os demais nós da rede (escravos) apenas respondem quando solicitados.
Rede de comunicação multimestre	Rede de comunicação onde às transferências de informações são iniciadas por qualquer nó ligado ao barramento de dados.
RX	Sigla usada para indicar recepção serial.
Sistema redundante	Sistema que contém elementos de reserva ou duplicados para executar determinada tarefa, que podem tolerar determinados tipos de falha sem que execução da tarefa seja comprometida.
SNTP	Simple Network Time Protocol. Protocolo para sincronização de tempo via rede.
SOE	Sequence Of Events. Serviço para monitorar a variação de entradas digitais pré-configuradas, salvando a data/horário da modificação e seu novo estado.
Software	Programas de computador, procedimentos e regras relacionadas à operação de um sistema de processamento de dados.
Subrede	Segmento de uma rede de comunicação que interliga um grupo de equipamentos (nós) com o objetivo de isolar o tráfego local ou utilizar diferentes protocolos ou meio físicos.
Tag	Nome associado a uma variável ou a uma lógica que permite uma identificação resumida de seu conteúdo.
Time-out	Tempo preestabelecido máximo para que uma comunicação seja completada. Se for excedido procedimentos de retentativa ou diagnósticos serão ativados.
Troca a quente	Procedimento de substituição de módulos de um sistema sem a necessidade de desenergização do mesmo. Normalmente utilizado em trocas de módulos de E/S.
TX	Sigla usada para indicar transmissão serial.
UCP	Sigla para unidade central de processamento. Controla o fluxo de informações, interpreta e executa as instruções do programa e monitora os dispositivos do sistema.
UCP ativa	Em um sistema redundante, a UCP ativa realiza o controle do sistema, lendo os valores dos pontos de entrada, executando o programa aplicativo e acionando os valores das saídas.
UCP inoperante	É a UCP que não está no estado ativo (controlando o sistema) nem no estado reserva (supervisionando a UCP ativa). Não pode assumir o controle do sistema.
UCP redundante	Corresponde à outra UCP do sistema, como, por exemplo, a UCP2 em relação à UCP1 e vice-versa.
UCP reserva	Em um sistema redundante, é a UCP que supervisiona a UCP ativa, não realizando o controle do sistema, mas estando pronta para assumir o controle em caso de falha na UCP ativa.
Upload	Leitura do programa ou configuração do CP.
Variável de Representação Direta	A variável pode ser acessada diretamente na memória, utilizando o endereço desejado. Por exemplo: %QB0, %MW100.
Variável Simbólica	Variáveis IEC criadas em POUs e GVLs durante o desenvolvimento do aplicativo, as quais não são endereçadas diretamente na memória.
WD	Sigla para cão-de-guarda em inglês (watchdog).
Word	Unidade de informação composta por 16 bits.
Tempo de execução	Veja Tempo de Ciclo.
Tempo de Ciclo	É o tempo que a UCP leva para executar uma determinada tarefa da aplicação.
Intervalo	Define de quanto em quanto tempo uma determinada tarefa será executada.
LCD	Sigla para Liquid Cristal Display ou Display de Cristal Líquido.
RS-232	É um padrão para troca serial de dados entre dois pontos (ponto a ponto)
RS-422	É um padrão para troca serial de dados entre dois ou mais pontos (ponto a ponto full duplex)

RS-485 É um padrão para troca serial de dados entre dois ou mais pontos (ponto a ponto half duplex).

10Base-T Tipo de nível físico para redes Ethernet, definida pela norma IEEE 802.3i em 1990. Suporta baud rates de 10 Mbps sobre dois pares de fios trançados de categoria 3.

Categoria 5 Uma das categorias do cabo UTP: par trançado não blindado com 100 ohms de impedância e características elétricas que suportam frequências de transmissão de até 100 MHz. Definido pela norma TIA/EIA 568-A, pode ser usado em redes 10Base-T e 100Base-TX, entre outras.

ScTP Sigla para screened twisted pair. Mesmo cabo que o UTP, porém todos os pares de fios estão envoltos por uma folha metálica, ou por uma tela metálica trançada, a fim de minimizar a radiação e a susceptibilidade ao ruído externo. Também é conhecido por sUTP (Screened Unshielded Twisted Pair) ou FTP (Foil Twisted Pair).

UTP Sigla para unshielded twisted pair. Tipo de cabo formado por pares não blindados de fios trançados. Para aplicações de rede, o termo UTP geralmente se refere ao cabo de 100 ohms, categoria 3, 4 ou 5, especificado pela norma TIA/EIA 568-A. Normalmente o cabo UTP possui 4 pares de fios trançados dentro da mesma bainha (encapsulamento externo).

Anexo A. Interoperabilidade DNP3

DNP3 Device Profile

DNP3 **DEVICE PROFILE DOCUMENT**	
Device Identification	
Vendor Name	Altus S/A
Device Name	NX3030
Device Function	Slave
DNP Levels Supported for	Requests: None Responses: None
Connections Supported	IP Networking
Methods to set Configurable Parameters	Software: Master Tool IEC XE
IP Networking	
Type of End Point:	TCP Listening (Outstation Only)
Accepts TCP Connections from	Allows all
IP Address(es) from which TCP Connections are accepted:	*.*.*.*
TCP Listen Port Number	Configurable, range 1to 65535
TCP Keep-alive timer	Configurable, range 0 to 4294967295
Multiple master connections	Supports up to two masters Based on TCP port number
Time synchronization support	SNTP
Link Layer	
Data Link Address	Configurable, range 0 to 65519
Self Address Support using address 0xFFFC	No
Requires Data Link Layer Confirmation	Never
Maximum number of octets Transmitted in a Data Link Frame	Fixed at 292
Maximum number of octets that can be Received in a Data Link Frame	Fixed at 292
Application Layer	
Maximum number of octets Transmitted in an Application Layer Fragment	Fixed at 2048
Maximum number of octets that can be received in an Application Layer Fragment	Fixed at 2048
Timeout waiting for Application Confirm of solicited response message	Fixed at 10000 ms
Device Trouble Bit IIN1.6	This bit will be set if PLC is not in RUN mode
Event Buffer Overflow Behavior	Discard the oldest event
Sends Multi-Fragment Responses	Yes
Outstation Unsolicited Response Support	
Supports Unsolicited Reporting	No

DNP V3.0 Implementation Table

DNP OBJECT GROUP & VARIATION			REQUEST Master may issue Outstation must parse		RESPONSE Master must parse Outstation may issue	
Group Num	**Var Num**	**Description**	**Function Codes (dec)**	**Qualifier Codes (hex)**	**Function Codes (dec)**	**Qualifier Codes (hex)**
1	0	Binary Input – Any Variation	1 (read)	00, 01 (start-stop) 06 (no range, or all)		
1	1	Binary Input – Packed format	1 (read)	00, 01 (start-stop) 06 (no range, or all)	129 (response)	00, 01 (start-stop)
2	0	Binary Input Event – Any Variation	1 (read)	06 (no range, or all) 07, 08 (limited qty)		
2	2	Binary Input Event – With absolute time	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response)	17, 28 (index)
60	1	Class Objects – Class 0 data	1 (read)	06 (no range,or all)		
60	2	Class Objects – Class 1 data	1 (read)	06 (no range, or all) 07, 08 (limited qty)		
80	1	Internal Indications – Packed format	1 (read)	00, 01 (start-stop)	129 (response)	00 (start-stop)
			2 (write)	00 (start-stop) index=7		